MATHEMATICS · LOGIC · PYTHON

Math & Logic Essentials for Programming

The same complete foundation in math & logic, with every idea bridged to Python – so you can start coding immediately.

LUKE MCLAUGHLIN



2026 Edition

LOGIC · ALGEBRA · NUMBER BASES · PYTHON

Math & Logic Essentials for Programming

2026 EDITION

Copyright © 2026 Luke McLaughlin. All rights reserved.

The moral right of the author to be identified as the author of this work has been asserted.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise) without the prior written permission of the author, except for brief quotations embodied in reviews and other non-commercial uses permitted by copyright law.

Written, designed, and **independently published by Luke McLaughlin**.

This book is intended for educational purposes. While every effort has been made to ensure the accuracy of its contents, the author assumes no responsibility for errors or omissions, or for any outcome arising from the use of the material herein.

Luke McLaughlin · Author & Publisher

lukekevinmclaughlin@gmail.com · lukekevinmclaughlin.com

First edition, 2026

Preface

This book exists for one reason: to make these subjects genuinely learnable by **anyone**. There is a stubborn myth that mathematics, logic, and programming belong only to people who are "naturally gifted" at them. They do not. They are learnable skills, and what they truly reward is not talent but *approach* and *persistence*. Given the right path and a little patience, anyone can master them.

The whole book is built on that belief. Every subject here is broken into small **building blocks** and arranged so that each one rests on the ones before it. You are never asked to take a leap; you take a step. Each chapter opens by explaining *why* an idea is true before showing *how* to use it, then backs it up with worked examples, mental-math shortcuts, the common mistakes to sidestep, and a short quiz so you can check yourself before moving on. In this edition every idea is also bridged to 🐍 **Python**: a "From Concept to Code" section in each chapter, plus a final hands-on programming part, so the moment you understand a concept you can watch it run as real code.

That structure is meant to serve two very different readers at once. If you are a **complete beginner**, you can start at page one and walk the entire path with nothing assumed; everything you need is built up from the ground. If you are a **professional** shoring up your foundations, the same building-block layout lets you drop in exactly where you need to and trust that the ground beneath it is solid.

Work through the parts in order, take the quizzes seriously, and give each block the time it needs. The goal is never to finish quickly; it is to truly *understand*. Do that, and these subjects will stop feeling like a wall and start feeling like a staircase.

About the Author



Luke McLaughlin is an independent author and publisher who grew up across several countries, an upbringing that left him endlessly curious and gave him a lifetime of overlapping passions. His path has wound through molecular biology, writing, digital growth, and software and artificial intelligence, including building *Codemonkey AI*, a desktop tutor that teaches people to code. The thread running through all of it is a love of taking something that looks complicated and uncovering the simple ideas underneath.

That is also his philosophy of learning. He treats every subject as a series of **building blocks**, each one resting on the one before it. Master the blocks in order and no subject is truly out of reach: give a little time, effort, and patience to each step, and even the most daunting topic gives way. This book was written in exactly that spirit.

> *"Every journey of a thousand miles begins with a single step."*

lukekevinmclaughlin@gmail.com · lukekevinmclaughlin.com

How to Use This Book

Every chapter follows the same rhythm. Look for these signposts:

- ***Definition*** – the precise meaning of a term or rule, in a tinted box.
- ***Example N → Solution*** – a worked problem with numbered steps. Try it yourself first, then check.
- ⚡ ***Mental-Math Shortcut*** – a faster way to get the answer in your head.
- ⚠ ***Common Mistakes*** – the traps most learners fall into, and how to avoid them.
- 📝 ***Quiz*** – a few questions to lock in the idea. Full answers appear at the end of each chapter.
- 🐍 ***From Concept to Code*** – the same idea written as Python you can run yourself.

Work through the parts in order – each one assumes the ones before it. Take the quizzes seriously; the goal is not to finish fast but to *understand*.

Contents

Part I – Number & Arithmetic Foundations

- Chapter 1 – Order of Operations (BODMAS / PEMDAS)
- Chapter 2 – Place Value & Comparing Numbers
- Chapter 3 – Addition & Subtraction
- Chapter 4 – Multiplication
- Chapter 5 – Division
- Chapter 6 – Integers & Signed Numbers

Part II – Factors, Multiples & Number Theory

- Chapter 7 – Factors, Primes & Factor Trees
- Chapter 8 – Greatest Common Factor & Least Common Multiple
- Chapter 9 – The Distributive Property

Part III – Exponents, Roots & Scientific Notation

- Chapter 10 – Exponents & the Laws of Powers
- Chapter 11 – Square Roots
- Chapter 12 – Scientific Notation
- Chapter 13 – Unit Conversions

Part IV – Fractions, Decimals & Percentages

- Chapter 14 – Fractions
- Chapter 15 – Decimals
- Chapter 16 – Percentages

Part V – Introduction to Logic

- Chapter 17 – Statements & Propositions
- Chapter 18 – Logical Connectives: AND, OR, NOT
- Chapter 19 – Truth Tables
- Chapter 20 – Conditionals & Implication
- Chapter 21 – Equivalence & De Morgan's Laws
- Chapter 22 – Quantifiers: For All & There Exists
- Chapter 23 – Sets & Venn Diagrams
- Chapter 24 – Boolean Algebra
- Chapter 25 – Logical Reasoning & Validity

Part VI – Algebra: Equations, Inequalities & Functions

- Chapter 26 – Variables & Expressions
- Chapter 27 – Solving Linear Equations
- Chapter 28 – Inequalities
- Chapter 29 – Functions & the Coordinate Plane

Part VII – Algebra in Depth

- Chapter 30 – Systems of Linear Equations
- Chapter 31 – Polynomials
- Chapter 32 – Exponentials & Logarithms
- Chapter 33 – Sequences, Series & Recurrence Relations
- Chapter 34 – Modular Arithmetic
- Chapter 35 – Sets, Relations & Mappings
- Chapter 36 – Matrices & Vectors

Part VIII – Trigonometry

- Chapter 37 – Angles & Radians
- Chapter 38 – Sine, Cosine & Tangent
- Chapter 39 – The Unit Circle
- Chapter 40 – Trigonometric Identities
- Chapter 41 – Inverse Trigonometric Functions
- Chapter 42 – Waves & Periodic Functions
- Chapter 43 – Rotations & Polar Coordinates

Part IX – Geometry

- Chapter 44 – Points, Lines & Planes
- Chapter 45 – Angles
- Chapter 46 – Triangles
- Chapter 47 – Circles
- Chapter 48 – Coordinate Geometry
- Chapter 49 – Distance, Area & Volume
- Chapter 50 – Transformations & Geometric Vectors
- Chapter 51 – Projections
- Chapter 52 – Intersection & Collision Geometry
- Chapter 53 – Graphs & Networks

Part X – Calculus

- Chapter 54 – Limits
- Chapter 55 – Derivatives
- Chapter 56 – Rates of Change & Motion
- Chapter 57 – Optimisation
- Chapter 58 – Integrals
- Chapter 59 – Multivariable Calculus: Partial Derivatives & Gradients
- Chapter 60 – Differential Equations
- Chapter 61 – Series & Approximations

Part XI – Number Bases & the Language of Computers

- Chapter 62 – Number Bases: A Primer

Chapter 63 – Base 10: The Decimal System
Chapter 64 – Base 2: Binary & How Computers Count

Part XII – Python: From Concept to Code

Chapter 65 – Getting Started: the REPL & print()
Chapter 66 – Variables & Types
Chapter 67 – Operators & Precedence
Chapter 68 – Booleans, Comparisons & Conditionals
Chapter 69 – Loops
Chapter 70 – Functions
Chapter 71 – Collections: Lists, Sets & Dictionaries
Chapter 72 – Putting It Together: Small Programs
Chapter 73 – Where to Go Next



PART I

Number & Arithmetic Foundations

How numbers are built and combined. The four operations, signed numbers, and the order that ties them together – plus mental-math shortcuts for doing it all in your head.

Chapter 1 – Order of Operations (BODMAS / PEMDAS) · Chapter 2 – Place Value & Comparing Numbers · Chapter 3 – Addition & Subtraction · Chapter 4 – Multiplication · Chapter 5 – Division · Chapter 6 – Integers & Signed Numbers

Chapter 1 · Order of Operations (BODMAS / PEMDAS)

How to evaluate an expression that mixes +, −, ×, ÷, brackets and powers – and always get the one correct answer.

The Idea Behind It

Why must $\times$ act before $+$? In $2 + 3 \times 4$, multiplying first gives 14, adding first gives 20. *but why?*

A page of symbols should mean **one** number to everyone who reads it, so we need a single agreed order. The order is not arbitrary: it follows from what multiplication *is*. The governing fact is that **multiplication is repeated addition**, 3×4 is just a shorthand for three 4s:

$$\boxed{3 \times 4 = 4 + 4 + 4}$$

So the 3×4 inside $2 + 3 \times 4$ is **one block**, the sum of three 4s. Replace it by what it stands for and the expression has only additions left, with no choice of order:

$$2 + \underbrace{3 \times 4}_{} = 2 + \underbrace{(4 + 4 + 4)}_{} = 2 + 4 + 4 + 4 = 14$$

Doing the $+$ first would glue the 2 onto **one** of those 4s before the block is complete, giving $5 \times 4 = 20$, the answer to a *different* question ($5 + 5 + 5 + 5$). The symbols never said that.

$$\therefore \text{ multiplication (the repeated operation) binds tighter than addition: } 2 + 3 \times 4 = 14$$

That is the whole reason for the order: $\times$ and $\div$ are built **on top of** $+$ and $-$, so they must be resolved first to keep every block intact. *build the blocks, then add them*

When an expression contains more than one operation, the order you work in changes the answer. Consider $3 + 4 \times 2$. If you add first you get 14; if you multiply first you get 11. Only one of these is correct, so mathematicians agreed on a fixed **order of operations** that everyone follows.

BODMAS / PEMDAS

Evaluate an expression in this order: - **B**rackets (also called **P**arentheses) - **O**rders – powers and roots (**E**xponents) - **D**ivision and **M**ultiplication – together, left to right - **A**ddition and **S**ubtraction – together, left to right

The two names describe the *same* rule. **BODMAS** is common in the UK; **PEMDAS** ("Please Excuse My Dear Aunt Sally") is common in the US. The single most important point:

$$\text{Multiplication and division rank } \textbf{equally}, \text{ as do addition and subtraction.}$$

When operations of equal rank sit side by side, you simply work **left to right**. So $20 \div 4 \times 2$ is $5 \times 2 = 10$, not $20 \div 8$.

In the worked examples below, the part being worked on **this step** is shaded, the **new value** it becomes is shaded a different colour, and the **final answer** is shaded green – so you can see exactly what each step touches.

Worked examples

Example 1

Evaluate $3 + 4 \times 2$.

SOLUTION

1. No brackets or powers, so the **multiplication** outranks the addition. *do × before +*
2. Multiply the shaded part: $4 \times 2 = 8$.
3. Now only addition is left: $3 + 8 = 11$. *just add*

$3 + 4 \times 2 = 11$

Example 2

Evaluate $2 + 80 \div (16 \div 2) \times 10$.

SOLUTION

1. **Brackets** first: $(16 \div 2) = 8$, giving $2 + 80 \div 8 \times 10$. *brackets win*
2. $\div$ and $\times$ are equal rank, so go **left to right**: $80 \div 8 = 10$.
3. Keep going left to right: $10 \times 10 = 100$, giving $2 + 100$. → *this way*
4. Finally the addition: $2 + 100 = 102$.

102

Example 3

Evaluate $5 + 2^3 \times (7 - 3)$.

SOLUTION

1. **Brackets**: $(7 - 3) = 4$.
2. **Orders** (the power): $2^3 = 8$. *2·2·2*
3. **Multiply**: $8 \times 4 = 32$.
4. **Add**: $5 + 32 = 37$.

37

Example 4

Evaluate $\dfrac{(6+2)^2}{4} - 3$.

SOLUTION

1. A fraction bar groups like a bracket, so finish the top first. Bracket: $(6+2) = 8$. *top first*
2. **Power**: $8^2 = 64$.
3. **Divide** by the bottom: $64 \div 4 = 16$.
4. **Subtract**: $16 - 3 = 13$.

13

Example 5

Evaluate $(-3)^2 + 12 \div 4 - 5$.

SOLUTION

1. **Bracket then power**: square the negative, $(-3)^2 = 9$. *− × − = +*
2. **Divide** before subtracting: $12 \div 4 = 3$, giving $9 + 3 - 5$.
3. Now equal-rank $+$ and $-$, **left to right**: $9 + 3 = 12$, then $12 - 5 = 7$.

7

Example 6

Evaluate $36 \div (2+1) \times 2 + 4$.

SOLUTION

1. **Brackets** first: $(2+1) = 3$, giving $36 \div 3 \times 2 + 4$. *brackets win*
2. $\div$ and $\times$ rank equally, so go **left to right** – divide before you multiply: $36 \div 3 = 12$. → *this way*
3. Still left to right: $12 \times 2 = 24$, giving $24 + 4$.
4. **Add** last: $24 + 4 = 28$.

28

Example 7

Evaluate $2 \times 3^2 + (10 - 4)^2 \div 9$.

SOLUTION

1. Two brackets/powers are in play; clear the **bracket** first: $(10 - 4) = 6$, so its square is 6^2.

   inside before the power

2. **Orders** (powers): $3^2 = 9$ and $6^2 = 36$, giving $2 \times 9 + 36 \div 9$.
3. **Multiply and divide**, each before any addition: $2 \times 9 = 18$ and $36 \div 9 = 4$.
4. **Add** what is left: $18 + 4 = 22$.

22

Example 8

Evaluate $\dfrac{20 - (3 + 1)^2}{2} + 7$.

SOLUTION

1. The fraction bar groups the whole top like a bracket, so finish the top before dividing. Inner **bracket**: $(3 + 1) = 4$, so the top is $20 - 4^2$. *top first*
2. **Power** on top: $4^2 = 16$, leaving $20 - 16$ above the bar.
3. **Subtract** to finish the top: $20 - 16 = 4$, so the fraction is $4/2$.
4. **Divide** by the bottom: $4 \div 2 = 2$, then **add**: $2 + 7 = 9$.

9

Note: A fraction bar, a square-root sign, and the inside of a power all act like invisible brackets: finish what is inside them before combining with the rest.

Mental-Math Shortcut

Read a long expression **left to right once**, but only *act* on the highest-ranking operation you can see, holding the rest in mind. For $100 - 6 \times 5$, your eye passes the $100-$, spots the $\times$, computes 30, then finishes $100 - 30 = 70$. With practice you evaluate the powers and products "in place" and sweep up the additions and subtractions last – no rewriting needed.

⚠ Common Mistakes

- Working strictly left to right regardless of rank – e.g. reading $3 + 4 \times 2$ as $7 \times 2 = 14$. Multiplication outranks addition.
- Thinking multiplication always comes before division (or addition before subtraction). They are **equal** rank – go left to right.
- Forgetting that a fraction bar or root sign groups everything beneath/within it like a bracket.
- Applying a power to the wrong base: $-3^2 = -(3^2) = -9$, but $(-3)^2 = 9$.

📝 Quiz

1.1.Q1. Evaluate $8 + 2 \times 5$.

A. 50

B. 18

C. 40

D. 15

1.1.Q2. Evaluate $24 \div 6 \times 2$.

A. 2

B. 8

C. 72

D. 12

1.1.Q3. Evaluate $(5 + 3)^2 - 10$.

1.1.Q4. Evaluate $3 \times (4 + 2^2)$.

1.1.Q5. True or False: in $12 - 4 + 3$, you must do the addition before the subtraction.

A. True

B. False

1.1.Q6. Evaluate $\dfrac{20 - 2 \times 4}{3}$.

🐍 From Concept to Code (Python)

Python obeys the **exact same order of operations** – it is built into the language as *operator precedence*. Type math straight into Python and it evaluates powers (`**`) first, then `* / // %`, then `+ −`, left to right, with parentheses overriding everything.

```
print(3 + 4 * 2)        # multiply first
print((3 + 4) * 2)      # parentheses force the add first
print(5 + 2 ** 3 * (7 - 3))
```

```
output
11
14
37
```

Try it:

1. Predict, then run: `print(20 / 4 * 2)`. Why is it `10.0` and not `2.5`?
2. Use parentheses to make `2 + 3 * 4` equal `20` instead of `14`.

Answer Key – Chapter 1 – Order of Operations (BODMAS / PEMDAS)

1.1.Q1. 18 – multiply first ($2 \times 5 = 10$), then add 8.

1.1.Q2. 8 – equal rank, left to right: $24 \div 6 = 4$, then $4 \times 2 = 8$.

1.1.Q3. 54 – bracket 8, square 64, subtract 10.

1.1.Q4. 24 – power 4, bracket $4 + 4 = 8$, times $3 = 24$.

1.1.Q5. False – equal rank, left to right: $12 - 4 = 8$, then $+3 = 11$.

1.1.Q6. 4 – top: $20 - 8 = 12$; then $12 \div 3 = 4$.

Chapter 2 · Place Value & Comparing Numbers

How to read, write and compare whole numbers (and the first decimals) by understanding what each digit is worth from its position.

The Idea Behind It

The same digit 3 is worth 30 in one column and 3 in another. Why does *position* change a digit's value?

but why?

Ten symbols (0 to 9) have to name *every* number, so we let a digit's **column** carry extra information: the worth of a column. The rule that powers everything is that **each place is worth ten times the place on its right**, climbing as powers of ten:

$$\boxed{\ldots,\ 10^2 = 100,\ 10^1 = 10,\ 10^0 = 1}$$

A digit's value is therefore **the digit multiplied by its place's power of ten**. Take 347 and read each column for what it is worth:

$$347 = \underbrace{3 \times 10^2}_{300} + \underbrace{4 \times 10^1}_{40} + \underbrace{7 \times 10^0}_{7} = 300 + 40 + 7 = 347$$

Now compare the lone 3 with the 3 in 30. They are the *same symbol*, but they sit in different columns, so they are multiplied by different powers of ten:

$$\underbrace{3}_{3\times10^0} = 3, \qquad \underbrace{30}_{3\times10^1} = 30, \qquad 30 = 10 \times 3$$

$$\therefore\ \text{a digit's worth} = \text{digit} \times (\text{power of ten for its place})$$

This is why a 0 is never "nothing": it is a place-holder that keeps the other digits in their proper columns, so $30 \neq 3$.

the column does the work, not the symbol

We write every number using just ten symbols, the **digits** $0, 1, 2, 3, 4, 5, 6, 7, 8, 9$. The secret to making ten symbols cover *every* number is **place value**: a digit's worth depends on *where* it sits. The 5 in 52 means fifty, while the 5 in 25 means five. Same symbol, ten times the value, because it stands one column further to the left.

PLACE VALUE

Our number system is **base ten**: each column is worth ten times the column to its right. Reading from the decimal point leftwards the columns are **ones**, **tens**, **hundreds**, **thousands**, and so on. Reading rightwards they are **tenths**, **hundredths**, **thousandths**.

So the number $3\,486$ is built from its columns:

$$3\,486 = 3{\times}1000\ +\ 4{\times}100\ +\ 8{\times}10\ +\ 6{\times}1$$

Writing a number out this way is its **expanded form**. The order matters enormously: a 0 is not "nothing" but a **place-holder** that keeps the other digits in their proper columns. Without it, 305 would collapse into 35.

Reading and writing numbers

To read a large number, split it into groups of three digits from the right: $7\,204\,915 \to 7 \mid 204 \mid 915$. Each group is read as a normal three-digit number followed by its label, **million, thousand,** then the bare ones. So $7\,204\,915$ is "seven million, two hundred four thousand, nine hundred fifteen". Going the other way, "nine thousand and forty" must reserve a column for the missing hundreds, giving 9040, not 940.

The number line

Every whole number has a home on the **number line**, a straight road with the small numbers on the left and the large ones on the right. Moving right means *more*; moving left means *less*. This single picture is the key to comparing and, later, to negatives and decimals.

Comparing numbers

COMPARISON SYMBOLS

- $a = b$ means a **equals** b (and $a \neq b$ means they are **not equal**).
- $a < b$ means a is **less than** b (it sits to the *left* on the number line).
- $a > b$ means a is **greater than** b.
- $a \leq b$ means a is **less than or equal to** b; $a \geq b$ means **greater than or equal to**.

The "open mouth" of $<$ and $>$ always gapes toward the **larger** number, like a hungry mouth choosing the bigger meal: $7 > 3$, and $3 < 7$.

To compare two whole numbers, you do **not** need to count. Compare the *leftmost* digits first: - The number with **more digits** is larger (assuming no leading zeros): $1000 > 999$. - If they have the same number of digits, scan **left to right** and stop at the first column where they differ.

Worked examples

Example 1

What is the place value of the 7 in $4\,728$?

SOLUTION

1. Label the columns from the right: 8 is ones, 2 is tens, 7 is hundreds, 4 is thousands. *right to left*
2. The 7 sits in the **hundreds** column.

It is worth $7 \times 100 = 700$**.**

Example 2

Write $60\,503$ in words and in expanded form.

SOLUTION

1. Group from the right: 60 | 503, i.e. sixty thousand and five hundred three. *split in 3s*
2. Note the 0 in the tens column is a place-holder, so it is "five hundred **three**", not "fifty-three".
   0 holds the column
3. Expanded: 6×10000 $+\,0\times1000+$ 5×100 $+\,0\times10+$ 3×1.

"Sixty thousand, five hundred three"; $60000+500+3$ **.**

Example 3

Compare $3\,481$ and $3\,479$ using $<$, $>$ or $=$.

SOLUTION

1. Both have four digits, so scan from the left. Thousands: $3=3$. Hundreds: $4=4$. *tie so far*
2. Tens differ: 8 versus 7. Since 8 $>$ 7, the first number is larger from this column on.
   first mismatch wins
3. No need to look at the ones.

$3\,481$ $>$ $3\,479$

Example 4

Put these in order from smallest to largest: $812,\ 821,\ 798,\ 8120$.

SOLUTION

1. 8120 has four digits, the others three, so it is the largest. *most digits = biggest*
2. Among the three-digit numbers compare leading digits: 798 starts with 7, the smallest hundreds digit, so it comes first.
3. For 812 and 821, hundreds tie at 8; tens decide: 1 $<$ 2, so $812<821$.

$798<812<821<8120$

Example 5

A first look at decimal places: which is larger, 0.7 or 0.65?

SOLUTION

1. Here 0.7 means 7 **tenths** and 0.65 means 6 tenths and 5 hundredths.
2. Compare the tenths column first: $7 > 6$, and that settles it, just as with whole numbers.

   tenths decide it
3. Length can mislead: 0.65 has more digits but is **smaller**, because the leftmost differing column wins.

   more digits ≠ bigger

$0.7 > 0.65$

Example 6

What is the place value of the 5 in $3\,254\,000$, and of the 4?

SOLUTION

1. Group from the right in threes: $3 \mid 254 \mid 000$. The right group is ones, the next is thousands, the next millions. *split in 3s*
2. Inside the thousands group 254, the digits are hundred-thousands, ten-thousands, thousands. So 2 is hundred-thousands, 5 is **ten-thousands**, 4 is **thousands**.
3. The 5 is worth $5 \times 10000 = 50000$; the 4 is worth $4 \times 1000 = 4000$. *position sets the value*

The 5 is worth 50000 and the 4 is worth 4000.

Example 7

Compare 0.5 and 0.500, then 0.48 and 0.6, using $<$, $>$ or $=$.

SOLUTION

1. Adding zeros to the **right** of a decimal changes nothing: pad to equal length, $0.5 = 0.5\,00$. Every column then matches 0.500. *trailing zeros are free*
2. For 0.48 versus 0.6, pad the shorter: $0.6 = 0.6\,0$. Now compare the tenths first: 4 versus 6.
3. Since $6 > 4$ in the tenths column, 0.6 is the larger – the hundredths never get a look-in.

   leftmost column wins

$0.5 = 0.500$ **and** $0.48 < 0.6$

Example 8

Put these decimals in order from smallest to largest: $0.3,\ 0.27,\ 0.305,\ 0.3$.

SOLUTION

1. Pad them all to three decimal places so the columns line up: $0.300,\ 0.270,\ 0.305,\ 0.300$.

   same length to compare

2. Compare the **tenths** first: 0.270 has a 2 where the others have 3, so 0.27 is the smallest.

   smallest tenths first

3. The remaining three all have 3 tenths and 0 hundredths; the thousandths decide: $0.300 = 0.300 < 0.305$.

   next column breaks the tie

4. So the two copies of 0.3 are equal, and 0.305 edges above them.

$$0.27 < 0.3 = 0.3 < 0.305$$

Note: More digits does not automatically mean a bigger number once a decimal point is involved. 0.9 beats 0.10, because 9 tenths is more than 1 tenth. Always compare column by column from the left, never by counting digits after the point.

digit place value expanded form number line place-holder

Mental-Math Shortcut

Compare from the front, not the back. To decide which of two numbers is larger you almost never need the whole number, just the first place where they differ.

1. **Count the digits.** Whole numbers with no leading zero: more digits = bigger. $10\,000 > 9\,999$ at a glance.
2. **Same length? Sweep left to right** and stop at the first mismatch. For $58\,341$ vs $58\,329$, the first three digits (583) match; the tens column (4 vs 2) decides it, so $58\,341$ is larger. You never look at the ones.
3. **Round to the leading digits to gauge size.** Treat $4\,812$ as "about $4\,800$" and $5\,103$ as "about $5\,100$" to see instantly which is bigger and roughly by how much.

This "front-end" habit is exactly how you skim a price tag or a scoreboard, your eye jumps to the leading digits and ignores the rest.

⚠ Common Mistakes

- Judging size by the *number of digits* after a decimal point: thinking $0.65 > 0.7$ because "$65 > 7$". Compare the tenths first; 0.7 is larger.
- Dropping a place-holder zero when writing from words: writing "three thousand and five" as 35 or 305 instead of 3005.
- Pointing the inequality the wrong way. The open mouth of $<$/$>$ faces the **bigger** number, so $4 < 9$ and $9 > 4$ – never $4 > 9$.
- Confusing the *digit* with its *value*: saying the 3 in $3\,000$ "is three" rather than three thousand.
- Treating $\leq$ as strictly less. $5 \leq 5$ is **true**, because $\leq$ allows equality.

📝 Quiz

2.1.Q1. What is the place value of the 9 in $29\,140$?

A. 9

B. 90

C. 900

D. 9000

2.1.Q2. Write $40\,072$ in words.

2.1.Q3. Fill the blank with $<$, $>$ or $=$: $6\,508 \;\square\; 6\,580$.

A. $<$

B. $>$

C. $=$

2.1.Q4. Which is larger, 0.4 or 0.39?

A. 0.4

B. 0.39

C. they are equal

2.1.Q5. Order from smallest to largest: $703,\ 730,\ 73,\ 7003$.

2.1.Q6. True or False: $12 \geq 12$.

A. True

B. False

2.1.Q7. In $58\,206$, which digit is in the hundreds column?

A. 5

B. 8

C. 2

D. 6

2.1.Q8. Write the expanded form of $9\,045$.

🐍 From Concept to Code (Python)

Python writes whole numbers as the type `int` – and crucially, it does **not** keep place-holder commas or leading zeros; you just type the digits. A **comparison** like < or == is a question, and Python answers it with the Boolean value `True` or `False`. (Use == to *test* equality; a single = *assigns* a value.)

```
x = 3481
y = 3479
print(x > y)        # is x bigger?
print(x == y)       # are they equal?
print(7 <= 7)       # less-than-OR-equal
```

```
output
True
False
True
```

```
# place value falls straight out of base ten
n = 4728
print(n // 100 % 10)   # the hundreds digit
print(n % 10)          # the ones digit
```

```
output
7
8
```

Try it:

1. Predict, then run: `print(0.7 > 0.65)`. Does Python agree with the lesson?
2. Set `a = 60503` and print its tens digit and its thousands digit using `//` and `%`.

Answer Key – Chapter 2 – Place Value & Comparing Numbers

2.1.Q1. 9000 – the 9 sits in the thousands column, so 9×1000.

2.1.Q2. "Forty thousand and seventy-two" – note the two place-holder zeros in the hundreds and (within the thousands group) positions.

2.1.Q3. $<$ – thousands and hundreds tie, then tens decide: $0 < 8$, so $6\,508 < 6\,580$.

2.1.Q4. 0.4 – tenths column: $4 > 3$, so 0.4 wins despite having fewer digits.

2.1.Q5. $73 < 703 < 730 < 7003$ – fewest digits first, then compare left to right.

2.1.Q6. True – $\geq$ means "greater than **or equal to**", and they are equal.

2.1.Q7. 2 – from the right: 6 ones, 0 tens, 2 hundreds.

2.1.Q8. $9000 + 0 + 40 + 5$ (i.e. $9{\times}1000 + 4{\times}10 + 5{\times}1$); the hundreds column is empty.

Chapter 3 · Addition & Subtraction

How to add and subtract multi-digit numbers reliably – carrying, borrowing, adding three numbers at once, and checking your answer with the inverse operation.

The Idea Behind It

When a column overflows we "carry the 1" into the next column. Why is that little 1 allowed?

but why?

Carrying looks like a trick, but it is forced by place value. The governing fact (Chapter 2) is that **ten ones make exactly one ten**, so a bundle of ten in any column can be re-labelled as a single unit in the column to its left:

$$\boxed{10 \text{ ones} = 1 \text{ ten}}$$

Take $27 + 15$. Split each number by place value and add the tens to the tens, the ones to the ones:

$$27 + 15 = (20 + 7) + (10 + 5) = (20 + 10) + (7 + 5) = 30 + 12$$

The ones column has overflowed to 12, which is too big for a single digit. But 12 ones is 1 ten and 2 ones, so trade that ten leftward, exactly the "carry":

$$30 + 12 = 30 + (10 + 2) = (30 + 10) + 2 = 40 + 2 = 42$$

$$\therefore \text{ carrying the 1 just trades 10 ones for 1 ten: } 27 + 15 = 42$$

Borrowing is the very same trade run backwards: unpack 1 ten into 10 ones when a column is too small to subtract.

carry = bundle up, borrow = unbundle

Addition combines amounts into a total; **subtraction** takes one amount away from another, or measures the gap between them. They are two sides of the same coin: every subtraction "$13 - 5 = 8$" is just an addition "$8 + 5 = 13$" read backwards. Holding both in mind at once is what lets you *check* your work, and it is the single best defence against silly slips.

ADDITION & SUBTRACTION

In a **sum** $a + b = c$, the numbers a and b are **addends** and c is the **total** (or sum). In a **difference** $a - b = c$, a is the **minuend**, b the **subtrahend**, and c the difference. Addition can be done in any order ($a + b = b + a$); subtraction cannot ($7 - 3 \neq 3 - 7$).

Lining up by place value

The written method stacks numbers so that **ones sit above ones, tens above tens**, and so on, then works **column by column from the right**. Place value (Chapter 2) is doing all the heavy lifting: you only ever add single digits, the columns keep track of size.

Carrying (regrouping in addition)

If a column totals 10 or more, it cannot fit in one digit, so you **carry**: write the ones digit and pass the ten into the next column left. Carrying is just trading "ten ones for one ten".

$$\begin{array}{r} 47 \\ +\,38 \\ \hline 85 \end{array} \qquad 7+8=15 \;\Rightarrow\; \text{write } 5,\ \text{carry } 1; \quad 1+4+3=8$$

Borrowing (regrouping in subtraction)

If a column's top digit is **too small** to subtract from, you **borrow**: take one unit from the column to the left (worth ten here) and add it on. Borrowing is carrying run in reverse, trading "one ten for ten ones".

Checking by the inverse

Because addition and subtraction undo each other, you can always verify: - check a subtraction $a-b=c$ by confirming $c+b=a$; - check an addition $a+b=c$ by confirming $c-b=a$.

Worked examples

Example 1

Add $476+248$.

SOLUTION

1. Ones: $6+8=14$. Write 4, carry 1. ← carry the 1
2. Tens: $1+7+4=12$. Write 2, carry 1. add the carried 1
3. Hundreds: $1+4+2=7$.

$476+248=724$

Example 2

Add three numbers: $35+47+28$.

SOLUTION

1. Ones column: $5+7+8=20$. Write 0, carry 2. ← carry the 2
2. Tens column: $2+3+4+2=11$. Write 11 as there is no further column.
3. Combine: the total is 110.

$35+47+28=110$

Example 3

Subtract $652 - 487$.

SOLUTION

1. Ones: $2 - 7$ will not go. Borrow from the tens: $12 - 7 = 5$. The tens digit drops from 5 to 4.

   borrow 10

2. Tens: $4 - 8$ will not go. Borrow from the hundreds: $14 - 8 = 6$. The hundreds digit drops from 6 to 5.

   borrow again

3. Hundreds: $5 - 4 = 1$.

$652 - 487 = 165$

Example 4

Subtract across a zero: $700 - 246$.

SOLUTION

1. Ones: $0 - 6$ will not go, but the tens are 0 too, so borrow from the hundreds first.

   zeros block the borrow

2. The 700 becomes 6 hundreds, 9 tens, 10 ones (you regroup once, cascading through the zeros).

   borrow cascades

3. Now subtract column by column: $10 - 6 = 4$, $9 - 4 = 5$, $6 - 2 = 4$.

$700 - 246 = 454$

Example 5

Check the result of Example 3 using the inverse.

SOLUTION

1. Example 3 claims $652 - 487 = 165$.
2. Add the answer back to the subtrahend: $165 + 487$. *undo with +*
3. Ones $5 + 7 = 12$ (carry 1); tens $1 + 6 + 8 = 15$ (carry 1); hundreds $1 + 1 + 4 = 6$. Total 652.
4. It matches the original minuend, so the subtraction is correct. *rebuilds the start*

Confirmed: $165 + 487 = 652$.

Example 6

A word problem. A library had $1\,204$ books, lent out 586, and later received a donation of 97. How many are on the shelves now?

SOLUTION

1. Start by removing the loans: $1\,204 - 586 = 618$. *lent out → subtract*
2. Then add the donation: $618 + 97 = 715$. *donation → add*

715 **books.**

Example 7

Add numbers of different lengths: $3\,845 + 67 + 9$.

SOLUTION

1. Line up by place value, so 67 and 9 tuck under the **right-hand** columns of $3\,845$. *right-align the ones*
2. Ones: $5 + 7 + 9 = 21$. Write 1, carry 2. *← carry the 2*
3. Tens: $2 + 4 + 6 = 12$. Write 2, carry 1. (Only $3\,845$ and 67 reach this column.)
4. Hundreds: $1 + 8 = 9$. Thousands: just the 3.

$3\,845 + 67 + 9 = 3\,921$

Example 8

Subtract across several zeros: $5\,000 - 1\,374$.

SOLUTION

1. Ones: $0 - 4$ will not go, and the tens and hundreds are 0 too, so the borrow must come from the thousands. *zeros block the borrow*
2. The $5\,000$ regroups to 4 thousands, 9 hundreds, 9 tens, 10 ones – one borrow cascading through all the zeros. *borrow cascades*
3. Now subtract column by column: $10 - 4 = 6$, $9 - 7 = 2$, $9 - 3 = 6$, $4 - 1 = 3$.
4. Check by the inverse: $3\,626 + 1\,374 = 5\,000$. *rebuilds the start*

$5\,000 - 1\,374 = 3\,626$

Example 9

A two-step money problem. A bank account holds €2\,450. A bill of €1\,675 is paid, then a salary of €1\,890 arrives. What is the balance?

SOLUTION

1. Pay the bill first – money out, so **subtract**: $2\,450 - 1\,675 = 775$. bill out → subtract
2. The salary is money in, so **add** it on: $775 + 1\,890 = 2\,665$. salary in → add
3. Quick check on the addition: $775 + 1\,890$ rounds to $800 + 1\,900 = 2\,700$, close to $2\,665$, so no slipped digit.

€ $2\,665$.

Note: Keep your columns straight. The commonest cause of a wrong answer is a misaligned digit – a stray ones digit sitting under the tens column. With pencil and paper, rule a light vertical line, or use squared paper, so every place stays in its lane.

addend sum difference carry borrow inverse operation

Mental-Math Shortcut

Add left to right; subtract by complement. On paper you work right to left, but in your head it is faster the other way, because you say the *big* part of the answer first and adjust.

Left-to-right addition (breakdown). Add the larger lumps first, then the smaller:

$$538 + 376 = 538 + 300 + 70 + 6 = 838 + 70 + 6 = 908 + 6 = 914.$$

Speak each running total "aloud" in your head so you don't lose it.

Adding by complement. When an addend is just under a round number, round it up and take the little bit back:

$$57 + 79 = 57 + (80 - 1) = 137 - 1 = 136.$$

Complement the number nearer the round ten or hundred:

$$267 + 385 = 267 + (400 - 15) = 667 - 15 = 652.$$

Subtracting by complement / adding-up. Subtracting a number close to a round one? Subtract the round number and add the difference back:

$$45 - 29 = 45 - (30 - 1) = 15 + 1 = 16, \qquad 524 - 386 = 524 - (400 - 14) = 124 + 14 = 138.$$

Or *count up* from the smaller to the larger, which is how a cashier gives change: from 29 up to 45 is 1 (to 30) then 15 (to 45), so 16.

⚠ Common Mistakes

- Forgetting to add the carried digit into the next column: doing $1 + 7 + 4$ as $7 + 4 = 11$ and losing the carried 1.
- Subtracting the smaller digit from the larger *within a column* regardless of which is on top – e.g. reading the ones of $652 - 487$ as $7 - 2 = 5$ instead of borrowing for $12 - 7$.
- Borrowing across a zero incorrectly: in $700 - 246$, forgetting that the borrow cascades, so the middle column is 9 (not 0 or 10).
- Misaligning the columns when the numbers have different lengths, so ones end up added to tens.
- Assuming subtraction is commutative: writing $3 - 7 = 4$. Order matters – $3 - 7$ is -4 (see Chapter 6), it is **not** $7 - 3$.

📝 Quiz

3.1.Q1. Add $389 + 476$.

3.1.Q2. Subtract $503 - 268$.

3.1.Q3. Add the three numbers $58 + 64 + 39$.

3.1.Q4. A subtraction gives $642 - 175 = 467$. Check it by the inverse – is it right?

A. Yes

B. No

3.1.Q5. Use the complement shortcut: $73 - 48$.

3.1.Q6. Use left-to-right addition: $254 + 835$.

3.1.Q7. True or False: subtraction is commutative, so $9 - 4 = 4 - 9$.

A. True

B. False

3.1.Q8. A till has €840. It pays out €265, then takes in €58. What is the new total?

🐍 From Concept to Code (Python)

Python adds with + and subtracts with −, in the order you'd expect. A very common pattern is updating a running total, and += is shorthand for "add this on": `total += 5` means `total = total + 5` (there is a matching −=).

```
a = 476
b = 248
print(a + b)        # a sum
print(a - b)        # a difference
```

```
output
724
228
```

```
total = 0
for price in [35, 47, 28]:   # add three numbers
    total += price            # running sum
print(total)
```

```
output
110
```

Try it:

1. Check a subtraction with the inverse: set `c = 652 - 487`, then print `c + 487` and confirm it equals 652.
2. Start `balance = 840`, then do `balance -= 265` and `balance += 58`. Print the result.

Answer Key – Chapter 3 – Addition & Subtraction

3.1.Q1. 865 – ones $9+6=15$ (carry 1), tens $1+8+7=16$ (carry 1), hundreds $1+3+4=8$.

3.1.Q2. 235 – borrow across the zero: $13-8=5$, $9-6=3$, $4-2=2$.

3.1.Q3. 161 – ones $8+4+9=21$ (carry 2), tens $2+5+6+3=16$.

3.1.Q4. Yes – $467 + 175 = 642$ matches the minuend, so it is correct.

3.1.Q5. 25 – $73 - (50 - 2) = 73 - 50 + 2 = 23 + 2 = 25$.

3.1.Q6. 1089 – $254 + 800 = 1054$, $+30 = 1084$, $+5 = 1089$.

3.1.Q7. False – $9 - 4 = 5$ but $4 - 9 = -5$; order matters.

3.1.Q8. €633 – $840 - 265 = 575$, then $575 + 58 = 633$.

Chapter 4 · Multiplication

How to multiply: from the logic of the times tables to multi-digit numbers (partial products and the standard algorithm) and quick multiplying by powers of ten.

The Idea Behind It

To do 23×7 we split it into 20×7 and 3×7 and add. Why may we break a number apart like that?

but why?

Long multiplication, grids and partial products are all the same one idea: chop the big number into its place-value pieces, multiply each piece, then add. What licenses the chop is the **distributive law**, multiplying a sum is the same as multiplying each part and adding:

$$\boxed{a \times (b + c) = a \times b + a \times c}$$

A number like 23 *is* a sum of place-value parts, $23 = 20 + 3$. Feed that into the law with 7 outside the bracket:

$$23 \times 7 = (20 + 3) \times 7 = 20 \times 7 + 3 \times 7$$

Each piece is now an easy product, one a times-table fact, the other a multiple of ten:

$$20 \times 7 + 3 \times 7 = 140 + 21 = 161$$

$$\therefore\ 23 \times 7 = 161\text{, and the columns of long multiplication are exactly this split by place value}$$

Each row of the column method (or each cell of a grid) is one of these partial products; the final addition just re-assembles them.

split by place value, multiply, add back

Multiplication is repeated addition. Writing 4×3 saves you from writing $3 + 3 + 3 + 3$; it means "four groups of three". That is the whole idea, and it instantly explains two facts you will lean on constantly: order does not matter, and a known product gives you the next one for free.

MULTIPLICATION

In $a \times b = c$, the numbers a and b are **factors** and c is the **product**. The sign may be written $\times$, $\cdot$, or (with letters) nothing at all: $3 \cdot 5$, 3×5 and $3a$ all denote multiplication.

Two properties make tables and mental work easy: - **Commutative:** $a \times b = b \times a$. A 3×8 rectangle has the same area turned on its side, so $3 \times 8 = 8 \times 3 = 24$. This *halves* the table you must learn. - **Each row builds on the last:** since $a \times (n+1) = a \times n + a$, once you know $6 \times 7 = 42$ you get $6 \times 8 = 42 + 6 = 48$.

Multiplying by powers of ten

Our base-ten system gives a beautiful shortcut. Multiplying a whole number by 10 shifts every digit one column to the left, which looks like **appending a zero**: $47 \times 10 = 470$. By 100, append two zeros; by 1000, three.

$$47 \times 100 = 4700, \qquad 36 \times 1000 = 36\,000$$

This also lets you break a big multiplication into easy pieces: $47 \times 30 = 47 \times 3 \times 10 = 141 \times 10 = 1410$.

Multi-digit multiplication: partial products

To multiply two multi-digit numbers, split one factor by place value and multiply each part separately, then add. These pieces are called **partial products**, and they are simply the **distributive property** (Chapter 9) in action:

$$23 \times 14 = 23 \times (10 + 4) = 23{\times}10 + 23{\times}4 = 230 + 92 = 322$$

The familiar **standard (column) algorithm** is the same calculation laid out compactly: multiply by the ones, then by the tens (writing a 0 place-holder because you are really multiplying by a *ten*), and add the rows.

Worked examples

Example 1

Compute 7×6 from a fact you already know.

SOLUTION

1. Suppose you remember $7 \times 5 = 35$.
2. Each extra group of 7 adds 7, so $7 \times 6 = 35 + 7 = 42$.

one more group of 7

$7 \times 6 = 42$

Example 2

Multiply 58×100.

SOLUTION

1. Multiplying by 100 shifts every digit two columns left.

×100 → 2 places

2. In writing, that appends two zeros: $58\,00$.

$58 \times 100 = 5800$

Example 3

Multiply 34×6 (single-digit multiplier).

SOLUTION

1. Split by place value: $34 = 30 + 4$.

split into tens + ones

2. Partial products: $30 \times 6 = 180$ and $4 \times 6 = 24$.
3. Add: $180 + 24 = 204$.

$34 \times 6 = 204$

Example 4

Multiply 23×14 using partial products.

SOLUTION

1. Break the 14: $23 \times 14 = 23 \times 10 + 23 \times 4$. *14 = 10 + 4*
2. $23 \times 10 = 230$.
3. $23 \times 4 = 92$.
4. Add the partial products: $230 + 92 = 322$.

$23 \times 14 = 322$

Example 5

Multiply 146×23 with the standard algorithm.

SOLUTION

1. Multiply by the ones digit 3: $146 \times 3 = 438$.
2. Multiply by the tens digit 2 (really 20), writing a place-holder 0: $146 \times 20 = 2920$.
   tens row → place-holder 0
3. Add the two rows: $438 + 2920 = 3358$.

$146 \times 23 = 3358$

Example 6

A word problem. A theatre has 24 rows of 32 seats. How many seats in total?

SOLUTION

1. Total seats $= 24 \times 32$. *rows × seats*
2. Partial products: $24 \times 30 = 720$ and $24 \times 2 = 48$.
3. Add: $720 + 48 = 768$.

768 **seats.**

Example 7

Multiply by a multiple of ten: 63×40.

SOLUTION

1. Split the round number into a digit times ten: $40 = 4 \times 10$. *40 = 4 × 10*
2. Multiply by the easy digit first: $63 \times 4 = 252$.
3. Now multiply by the 10, which just appends one zero: $252 \times 10 = 252\,0$. *×10 → append a zero*

$63 \times 40 = 2520$

Example 8

Multiply 345×27 with the standard algorithm.

SOLUTION

1. Multiply by the ones digit 7: $345 \times 7 = 2415$ (carries: $7\times5=35$, $7\times4+3=31$, $7\times3+3=24$).
   carry as you go
2. Multiply by the tens digit 2 (really 20), writing a place-holder 0: $345 \times 20 = 6900$.
   tens row → place-holder 0
3. Add the two rows: $2415 + 6900 = 9315$.

$345 \times 27 = 9315$

Example 9

A money word problem. A shop sells 125 tickets at €18 each. What is the total revenue?

SOLUTION

1. Total $= 125 \times 18$. *tickets × price*
2. Break the price into tens and ones: $18 = 10 + 8$.
3. Partial products: $125 \times 10 = 1250$ and $125 \times 8 = 1000$.
4. Add: $1250 + 1000 = 2250$. *sum the parts*

€ 2250.

Note: When you multiply by the tens digit in the column algorithm, the place-holder 0 is essential – you are multiplying by a *ten*, not a one. Skip it and every product from the tens row lands in the wrong column.

factor product commutative partial product power of ten

⚡ Mental-Math Shortcut

Almost every fast product comes from **doubling, "by the previous", or a round-number trick**.

×2, ×3, ×4 by the previous. Doubling is adding a number to itself ($8 \times 2 = 8 + 8 = 16$). Then $\times 3$ is "double plus one more": $8 \times 3 = 8{\times}2 + 8 = 16 + 8 = 24$. And $\times 4$ is **double twice**: $8 \times 4 = (8{\times}2){\times}2 = 16{\times}2 = 32$.

×5 = halve, then ×10. Because $5 = 10 \div 2$: $6 \times 5 = (6 \div 2) \times 10 = 30$. For an odd number go via the previous: $7 \times 5 = 6 \times 5 + 5 = 35$.

×9 = ×10 minus the number. $7 \times 9 = 70 - 7 = 63$.

×11 (two digits) = spread and insert the sum. Split the digits and drop their sum between them: $53 \times 11 = 5\,|\,(5{+}3)\,|\,3 = 583$. If the middle sum reaches ten, carry it: $84 \times 11 = 8\,|\,12\,|\,4 = 924$.

×25 = ×100 ÷ 4. $24 \times 25 = 2400 \div 4 = 600$. Not divisible by 4? Use a complement: $57 \times 25 = (60 - 3) \times 25 = 1500 - 75 = 1425$.

×50 = ×100 ÷ 2. $36 \times 50 = 3600 \div 2 = 1800$.

Squaring a number ending in 5. Take the tens digit n, compute $n(n{+}1)$, and append 25: $35^2 \to 3{\times}4 = 12$, then "25", giving 1225. Likewise $25^2 = (2{\times}3)25 = 625$.

Doubling and halving a product. You may double one factor and halve the other without changing the answer: $14 \times 5 = 7 \times 10 = 70$, and $16 \times 25 = 8 \times 50 = 4 \times 100 = 400$.

⚠ Common Mistakes

- Dropping the place-holder zero in the column method, e.g. adding 146×3 and 146×2 instead of 146×20 – the tens row must be shifted one place left.
- Adding zeros for the *wrong* power of ten: 58×1000 needs **three** zeros ($58\,000$), not two.
- Forgetting to carry between partial digits, e.g. $7 \times 8 = 56$ but writing 6 and not carrying the 5.
- Misremembering the ×11 trick when the inner sum exceeds nine: 84×11 is 924 (carry the 1 from $8{+}4{=}12$), not 8124.
- For squaring-ending-in-5, multiplying the tens digit by itself instead of by the **next** number: 35^2 uses $3 \times 4 = 12$, not $3 \times 3 = 9$.

Quiz

4.1.Q1. Compute 8×7.

A. 54

B. 56

C. 48

D. 64

4.1.Q2. Multiply 63×100.

4.1.Q3. Multiply 47×6 using partial products.

4.1.Q4. Multiply 34×12.

4.1.Q5. Use the ×11 shortcut: 72×11.

4.1.Q6. Use squaring-ending-in-5: 45^2.

4.1.Q7. True or False: $6 \times 9 = 9 \times 6$.

A. True

B. False

4.1.Q8. A box holds 16 rows of 25 tiles. How many tiles?

🐍 From Concept to Code (Python)

Python multiplies with $*$. It also has $**$ for **powers** (repeated multiplication, the subject of Chapter 10): `2 ** 5` is $2^5 = 32$. A neat extra: $*$ works on **strings** too, where it repeats them – a literal echo of "multiplication is repeated addition".

```
print(23 * 14)       # a product
print(58 * 100)      # times a power of ten
print(2 ** 5)        # 2 to the power 5
```

```
output
322
5800
32
```

```
print("ab" * 3)      # repeat a string
print("-" * 10)      # a quick divider line
```

```
output
ababab
----------
```

Try it:

1. Predict, then run: `print(34 * 12)` and check it equals the worked example.
2. Print a row of 20 stars without typing 20 of them, using `"*" * 20`.

Answer Key – Chapter 4 – Multiplication

4.1.Q1. 56 – e.g. $8 \times 7 = 8 \times 6 + 8 = 48 + 8 = 56$.

4.1.Q2. 6300 – append two zeros when multiplying by 100.

4.1.Q3. 282 – $40 \times 6 = 240$ and $7 \times 6 = 42$; $240 + 42 = 282$.

4.1.Q4. 408 – $34 \times 10 = 340$, $34 \times 2 = 68$; $340 + 68 = 408$.

4.1.Q5. 792 – spread $7 \mid (7+2) = 9 \mid 2$, giving 792.

4.1.Q6. 2025 – $4 \times 5 = 20$, then append 25: 2025.

4.1.Q7. True – multiplication is commutative; both equal 54.

4.1.Q8. 400 – halve/double: $16 \times 25 = 8 \times 50 = 4 \times 100 = 400$.

Chapter 5 · Division

How to divide: long division, the quotient and remainder, dividing by powers of ten, and how division simply undoes multiplication.

The Idea Behind It

Dividing by 0 is "undefined", not just hard. Why can $a \div 0$ have no answer at all? *but why?*

Division is not a new operation, it is a **question about multiplication**. The governing definition is that $a \div b$ asks for the missing factor: the number that multiplies b back up to a.

$$\boxed{a \div b = c \iff c \times b = a}$$

That is why division undoes multiplication: $12 \div 3 = 4$ holds precisely because $4 \times 3 = 12$. Now point the same question at $b = 0$ and ask for $a \div 0$ with $a \neq 0$, say $5 \div 0$. By the definition we need a number c with:

$$c \times 0 = 5$$

But **anything** times 0 is 0, never 5, so the left side is stuck at 0 whatever c we try:

$$c \times 0 = 0 \neq 5 \quad \text{for every } c$$

$$\therefore \text{ no } c \text{ exists, so } 5 \div 0 \text{ is } \text{undefined}$$

There is simply no missing factor to find: 0 erases every number it multiplies, so it can never be un-multiplied.

you cannot undo what wipes everything to zero

Division splits a quantity into equal parts – or asks "how many of these fit into that?". Sharing 12 sweets among 3 children gives $12 \div 3 = 4$ each. It is the exact reverse of multiplication: because $4 \times 3 = 12$, we have $12 \div 3 = 4$. Keeping that partnership in view lets you check every division by multiplying back.

DIVISION

In $a \div b = c$, the number a is the **dividend**, b the **divisor**, and c the **quotient**. By definition $a \div b = c$ exactly when $c \times b = a$. You may write it $a \div b$, a/b, or $\frac{a}{b}$. Dividing by **zero is undefined**: no number times 0 gives a non-zero a.

Quotient and remainder

Numbers do not always divide evenly. $17 \div 5$ gives 3 with 2 left over: three whole fives (15) fit, and 2 remains. We call 3 the **quotient** and 2 the **remainder**, and write the relationship as

$$a = q \times b + r, \qquad 0 \leq r < b \quad (\text{here } 17 = 3 \times 5 + 2).$$

Long division

For larger numbers we use **long division**, working through the dividend one digit at a time, left to right. At each step ask "how many times does the divisor go in?", write that digit of the quotient, subtract, and bring down the next digit. The leftover at the very end is the remainder.

Dividing by powers of ten

Mirroring multiplication, dividing a whole number by 10 shifts digits one column to the right. When the number ends in zeros this looks like **removing a zero**: $4700 \div 100 = 47$. If it does not divide evenly, the leftover digits move past the decimal point: $47 \div 10 = 4.7$.

Worked examples

Example 1

Compute $56 \div 8$ using the times tables.

SOLUTION

1. Ask: what number times 8 gives 56? ÷ undoes ×
2. From the eight times table, $7 \times 8 = 56$.

$56 \div 8 = 7$

Example 2

Divide with a remainder: $47 \div 6$.

SOLUTION

1. Find the largest multiple of 6 not exceeding 47: $6 \times 7 = 42$ (while $6 \times 8 = 48$ is too big). biggest that fits
2. The quotient is 7; the remainder is $47 - 42 = 5$. what is left over
3. Check: $7 \times 6 + 5 = 42 + 5 = 47$. rebuild the dividend

$47 \div 6 = 7$ **remainder** 5

Example 3

Divide by a power of ten: $8400 \div 100$.

SOLUTION

1. Dividing by 100 shifts every digit two columns right. ÷100 → 2 places
2. Since the number ends in two zeros, this removes them cleanly: $84\,00$.

$8400 \div 100 = 84$

Example 4

Long division: $864 \div 4$.

SOLUTION

1. Hundreds: $8 \div 4 = 2$, exactly. Write 2; nothing carried. *left digit first*
2. Tens: bring down the 6. $6 \div 4 = 1$ remainder 2. Write 1, carry the 2 to make 26... here the 2 tens stay as 20, so bring down next. *carry the leftover*
3. Ones: $20 + 4$ gives the 24 to share; $24 \div 4 = 6$. Write 6.
4. No remainder.

$864 \div 4 = 216$

Example 5

Long division with a remainder: $947 \div 7$.

SOLUTION

1. Hundreds: $9 \div 7 = 1$ remainder 2. Write 1. *carry the 2*
2. Tens: bring down the 4 to make 24. $24 \div 7 = 3$ remainder 3. Write 3. *bring down next*
3. Ones: bring down the 7 to make 37. $37 \div 7 = 5$ remainder 2. Write 5.
4. Check: $135 \times 7 + 2 = 945 + 2 = 947$. *multiply back to verify*

$947 \div 7 = 135$ **remainder** 2

Example 6

A word problem. 150 chairs are arranged into rows of 12. How many full rows, and how many chairs are left over?

SOLUTION

1. Divide: $150 \div 12$. *chairs ÷ row size*
2. $12 \times 12 = 144$, and $12 \times 13 = 156$ is too many, so the quotient is 12. *biggest that fits*
3. Remainder: $150 - 144 = 6$.

12 **full rows with** 6 **chairs left over.**

Example 7

Long division with a zero in the quotient: $824 \div 4$.

SOLUTION

1. Hundreds: $8 \div 4 = 2$, exactly. Write 2. *left digit first*
2. Tens: bring down the 2. But 4 does **not** go into 2, so write a 0 in the quotient and carry the 2.
   4 will not fit → write 0
3. Ones: bring down the 4 to make 24. $24 \div 4 = 6$. Write 6.
4. Check: $206 \times 4 = 824$. *multiply back to verify*

$824 \div 4 = 206$

Example 8

Long division by a two-digit divisor: $682 \div 22$.

SOLUTION

1. Look at the first two digits, 68: how many 22s fit? $22 \times 3 = 66$ (while $22 \times 4 = 88$ is too big), so the first quotient digit is 3. *biggest that fits*
2. Subtract and bring down: $68 - 66 = 2$, bring down the 2 to make 22. *bring down next*
3. How many 22s in 22? Exactly $22 \div 22 = 1$. Write 1, nothing left over.
4. Check: $31 \times 22 = 682$. *multiply back to verify*

$682 \div 22 = 31$

Example 9

A remainder you must interpret. 100 people need taxis that each seat 7. How many taxis are needed?

SOLUTION

1. Divide: $100 \div 7$. *people ÷ seats*
2. $7 \times 14 = 98$, and $7 \times 15 = 105$ is too many, so the quotient is 14 with remainder $100 - 98 = 2$.
   biggest that fits
3. Those leftover 2 people still need a taxi, so **round up**: $14 + 1 = 15$. *leftovers need 1 more*

15 **taxis (the last one only part-full).**

Note: Always sanity-check a division by multiplying back: quotient × divisor + remainder should rebuild the dividend. It is the fastest way to catch a slipped digit in long division.

dividend divisor quotient remainder divisible

⚡ Mental-Math Shortcut

Two big ideas: **test divisibility instantly**, and **divide by complement or breakdown**.

Divisibility rules (does b divide evenly?): - **2** – last digit is even $(0, 2, 4, 6, 8)$. - **5** – last digit is 0 or 5. **10** – last digit is 0. - **3** – the **digit sum** is divisible by 3 (e.g. $471 \to 4+7+1 = 12$, yes). - **9** – the digit sum is divisible by 9 $(729 \to 18$, yes). - **6** – divisible by **both** 2 and 3. - **4** – the **last two digits** form a number divisible by 4 $(316 \to 16$, yes). - **8** – the **last three digits** are divisible by 8 $(1\,720 \to 720 = 8\times 90$, yes). - **11** – the alternating digit sum is 0 or a multiple of 11 $(2\,728 \to 2-7+2-8 = -11$, yes).

Division by complement. Round the dividend *up* to a near multiple of the divisor, then take the bit back:

$$87 \div 3 = (90 - 3) \div 3 = 30 - 1 = 29, \qquad 342 \div 6 = (360 - 18) \div 6 = 60 - 3 = 57.$$

Division by breakdown. Split the dividend into easy multiples of the divisor and divide each:

$$72 \div 3 = (60 + 12) \div 3 = 20 + 4 = 24, \qquad 536 \div 8 = (400 + 136) \div 8 = 50 + 17 = 67.$$

⚠ Common Mistakes

- Reversing dividend and divisor: division is **not** commutative, so $12 \div 3 = 4$ but $3 \div 12 = 0.25$.
- Trying to divide by zero. $a \div 0$ has no answer – it is undefined, not 0 and not a.
- Letting the remainder be as large as (or larger than) the divisor: a remainder must satisfy $0 \le r < b$. If you get $r = 6$ when dividing by 5, the quotient was too small.
- In long division, forgetting to write a 0 in the quotient when the divisor will not go into a brought-down value (e.g. $618 \div 6$ has a 0 in the tens place: 103).
- Removing the wrong number of zeros for the power of ten: $8400 \div 1000 = 8.4$, not 84.

Quiz

5.1.Q1. Compute $72 \div 9$.

A. 6

B. 7

C. 8

D. 9

5.1.Q2. Divide $59 \div 6$, giving quotient and remainder.

5.1.Q3. Compute $7300 \div 100$.

5.1.Q4. Long division: $738 \div 6$.

5.1.Q5. Is 471 divisible by 3?

A. Yes

B. No

5.1.Q6. Use the complement shortcut: $342 \div 6$.

5.1.Q7. True or False: $0 \div 9 = 0$.

A. True

B. False

5.1.Q8. 130 apples are packed 8 to a bag. How many full bags, and how many left over?

🐍 From Concept to Code (Python)

Python gives you three related operators. `/` is **true division** and always returns a float (`17 / 5` → `3.4`). `//` is **floor division** – the whole-number quotient (`17 // 5` → 3). `%` (modulo) gives the **remainder** (`17 % 5` → 2). The built-in `divmod(a, b)` hands you the quotient *and* remainder together as a pair.

```
print(17 / 5)        # true division -> float
print(17 // 5)       # quotient (floor)
print(17 % 5)        # remainder
print(divmod(17, 5)) # both at once
```

```
output
3.4
3
2
(3, 2)
```

```
# divisibility test with %
n = 471
print(n % 3 == 0)    # divisible by 3?
print(150 % 12)      # leftover chairs
```

```
output
True
6
```

Try it:

1. Predict, then run: `print(8400 // 100)` and `print(8400 % 100)`.
2. Use `divmod(947, 7)` and confirm the quotient and remainder match the worked example.

Answer Key – Chapter 5 – Division

5.1.Q1. 8 – because $8 \times 9 = 72$.

5.1.Q2. 9 remainder 5 – $6 \times 9 = 54$, and $59 - 54 = 5$.

5.1.Q3. 73 – dividing by 100 removes two trailing zeros.

5.1.Q4. 123 – 6 into 7 is 1 r1; into 13 is 2 r1; into 18 is 3. Check: $123 \times 6 = 738$.

5.1.Q5. Yes – digit sum $4+7+1 = 12$, which is divisible by 3.

5.1.Q6. 57 – $(360 - 18) \div 6 = 60 - 3 = 57$.

5.1.Q7. True – zero shared into any (non-zero) number of parts is 0. (But $9 \div 0$ is undefined.)

5.1.Q8. 16 bags, 2 left – $8 \times 16 = 128$, remainder 2.

Chapter 6 · Integers & Signed Numbers

How to work with negative numbers: the extended number line, absolute value, and the sign rules for adding, subtracting, multiplying and dividing signed numbers.

The Idea Behind It

A minus times a minus comes out positive: $(-1) \times (-1) = +1$. *but why?*

A negative sign means *"the opposite direction."* Multiplying by -1 turns a number around to face the other way: turn around **once** and you point backwards, turn around **twice** and you face forward again. Two flips cancel. Here is that same fact forced by a rule you already trust, the **distributive law** $a(b+c) = ab + ac$:

$$(-1)\big(\,1 + (-1)\,\big) = (-1)(1) + (-1)(-1)$$

The bracket on the left is just $1 + (-1) = 0$, and anything times 0 is 0, so the whole left side is 0:

$$0 = \underbrace{(-1)(1)}_{=-1} + (-1)(-1)$$

For the right side to equal 0, the term $(-1)(-1)$ must be exactly what cancels the -1. The only number that does that is $+1$. *it has to undo the -1*

$$\therefore\ (-1)(-1) = +1$$

Run the same argument with any sizes and you get the rule for *every* product and quotient: **same signs make a positive, different signs make a negative.**

Some quantities go *below* zero: a temperature of $-5°$, a bank balance of -30 euros (an overdraft), a basement on floor -2. To describe them we extend the counting numbers leftwards past zero into the **negative** numbers. Together with zero and the positives, these form the **integers**.

INTEGERS

The **integers** are the whole numbers together with their negatives:

$$\ldots, -3, -2, -1,\ 0,\ 1, 2, 3, \ldots$$

written $\mathbb{Z}$. A number greater than 0 is **positive**; a number less than 0 is **negative**; **zero** is neither. The "$-$" sign in -3 marks direction, not subtraction.

The extended number line

Picture the number line stretching both ways from zero: positives to the right, negatives to the left, a mirror image. Everything from Chapter 2 still holds, **further right is larger**. So $-1 > -5$, because -1 sits to the right of -5. (A common surprise: among negatives, the one that *looks* bigger is actually smaller.)

$$-5 < -2 < 0 < 3 < 7$$

Absolute value

ABSOLUTE VALUE

The **absolute value** $|a|$ is the distance of a from zero on the number line, ignoring direction. Distance is never negative, so $|-4| = 4$ and $|4| = 4$ and $|0| = 0$.

Adding signed numbers

Think of moving along the number line: a positive number steps **right**, a negative number steps **left**. - **Same signs:** add the absolute values and keep the shared sign. $(-3) + (-5) = -8$ (two steps left, then more left). - **Different signs:** subtract the smaller absolute value from the larger, and take the sign of the number with the **larger** absolute value. $(-7) + 3 = -4;\ 7 + (-3) = 4$.

Subtracting signed numbers

Subtracting is adding the opposite. To subtract a number, flip its sign and add:

$$a - b = a + (-b), \qquad \text{so} \quad 5 - (-2) = 5 + 2 = 7.$$

The two minus signs in $5 - (-2)$ "cancel": taking away a debt of 2 leaves you 2 better off.

Multiplying and dividing signed numbers

Here the rule is wonderfully simple and the **same for** $\times$ **and** $\div$: combine the digits as usual, then fix the sign by counting minus signs. - **Same signs → positive:** $(-) \times (-) = (+)$ and $(+) \times (+) = (+)$. - **Different signs → negative:** $(+) \times (-) = (-)$ and $(-) \times (+) = (-)$.

$$(-) \times (-) = (+), \quad (-) \times (+) = (-), \quad (+) \times (-) = (-), \quad (+) \times (+) = (+)$$

Worked examples

Example 1

Order from least to greatest: $3,\ -6,\ 0,\ -1,\ 4$.

SOLUTION

1. Negatives sit left of zero; the further left, the smaller, so $-6 < -1$. further left = smaller
2. Then zero, then the positives in their usual order.

$-6 < -1 < 0 < 3 < 4$

Example 2

Evaluate $|-9| + |4|$.

SOLUTION

1. Absolute value is distance from zero: $|-9| = 9$ and $|4| = 4$. *distance ≥ 0*
2. Add the (now positive) results: $9 + 4 = 13$.

13

Example 3

Add $(-7) + (-5)$.

SOLUTION

1. Same signs, so add the absolute values: $7 + 5 = 12$. *same sign → add*
2. Keep the shared sign (negative).

$(-7) + (-5) = -12$

Example 4

Add $(-8) + 3$.

SOLUTION

1. Different signs, so subtract the smaller absolute value from the larger: $8 - 3 = 5$.

   different signs → subtract
2. Take the sign of the number with the larger absolute value, which is -8 (negative).

   bigger size wins the sign

$(-8) + 3 = -5$

Example 5

Subtract $4 - (-6)$.

SOLUTION

1. Subtracting is adding the opposite: $4 - (-6) = 4 + 6$. *two minuses → plus*
2. Add: $4 + 6 = 10$.

$4 - (-6) = 10$

Example 6

Multiply and divide: (a) $(-6) \times (-4)$ and (b) $(-20) \div 5$.

SOLUTION

1. (a) Same signs give a positive product: $6 \times 4 = 24$, so the answer is $+24$. − × − = +
2. (b) Different signs give a negative quotient: $20 \div 5 = 4$, so the answer is -4. − ÷ + = −

(a) 24 **(b)** -4

Example 7

A combined problem. The temperature is -3° at dawn, rises 9° by noon, then drops 5° by night. What is the night temperature?

SOLUTION

1. Start at -3, then add 9: $(-3) + 9 = 6$ (different signs, $9 - 3 = 6$, positive). rises → add
2. Then subtract 5: $6 - 5 = 1$. drops → subtract

1°

Example 8

Subtract $-4 - 7$.

SOLUTION

1. Turn the subtraction into adding the opposite: $-4 \; -7 = -4 \; +(-7)$. subtract → add the opposite
2. Now both signs are the same, so add the absolute values: $4 + 7 = 11$, and keep the negative sign.
   same sign → add

$-4 - 7 = -11$

Example 9

Multiply a chain: $(-2) \times (-3) \times (-5)$.

SOLUTION

1. Combine the digits as usual: $2 \times 3 \times 5 = 30$. The size of the answer is 30 whatever the signs.
   sizes multiply normally
2. Now fix the sign by **counting the minus signs**: there are 3 of them. count the minuses
3. An **odd** count of negatives gives a negative result. odd → negative

$(-2) \times (-3) \times (-5) = -30$

Example 10

Order of operations with signs: $-3 \times (2-6) + 10 \div (-5)$.

SOLUTION

1. **Brackets** first: $(2-6) = -4$, giving $-3 \times (-4) + 10 \div (-5)$. brackets win
2. **Multiply and divide** before adding. Same signs multiply to a positive: $(-3) \times (-4) = 12$. − × − = +
3. Different signs divide to a negative: $10 \div (-5) = -2$. + ÷ − = −
4. **Add** what is left: $12 + (-2) = 10$.

10

Note: A minus sign attached to a number (its sign) and a minus sign meaning "subtract" are written the same way but mean different things. Brackets keep them apart: $5 - (-2)$ reads "five, subtract negative two".

integer negative number absolute value opposite sign

Mental-Math Shortcut

The whole of signed arithmetic collapses to **two tiny rules** plus one rewrite.

For $\times$ and $\div$ – count the minus signs: - **Same sign → positive**, **different signs → negative**. So $(-)(-) = (+)$ and $(-)(+) = (-)$. - Shortcut for a chain: an **even** number of negative factors gives a positive result, an **odd** number gives a negative. $(-2)(-3)(-1) = -6$ (three minuses → negative).

For $+$ and $-$ – same sign add, different sign subtract: - Same sign: add the sizes, keep the sign. $(-6) + (-7) = -13$. - Different sign: subtract the smaller size from the larger, take the sign of the bigger. $(-6) + 7 = +1$; $6 + (-7) = -1$.

Turn every subtraction into an addition: $a - b = a + (-b)$. Then you only ever *add*, which means you only ever use the two rules above. $5 - (-2) = 5 + 2 = 7$, and $-4 - 3 = -4 + (-3) = -7$.

⚠ Common Mistakes

- Thinking -1 is smaller than -5 because "5 is bigger". On the number line -1 is to the **right** of -5, so $-1 > -5$.
- Letting absolute value return a negative: $|-9| = 9$, never -9. Distance cannot be negative.
- Mishandling double negatives in subtraction: $7 - (-3)$ is $7 + 3 = 10$, **not** 4.
- Confusing the addition rule with the multiplication rule: $(-3) + (-5) = -8$ (add, keep sign), but $(-3) \times (-5) = +15$ (two negatives make a positive). Adding two negatives does **not** give a positive.
- Forgetting the sign on a product or quotient entirely, e.g. writing $(-6) \times 4 = 24$ instead of -24.

Quiz

6.1.Q1. Which is larger, -2 or -9?

A. -2

B. -9

C. they are equal

6.1.Q2. Evaluate $|-12|$.

A. -12

B. 12

C. 0

6.1.Q3. Compute $(-8) + (-5)$.

6.1.Q4. Compute $(-9) + 4$.

6.1.Q5. Compute $6 - (-7)$.

6.1.Q6. Compute $(-7) \times (-6)$.

A. -42

B. 42

C. -13

D. 13

6.1.Q7. Compute $(-24) \div 6$.

6.1.Q8. True or False: the product $(-2)(-3)(-4)$ is positive.

A. True

B. False

6.1.Q9. The temperature is -4° and falls another 3°. What is it now?

🐍 From Concept to Code (Python)

Python writes a negative number with a leading minus, e.g. `−5`, and obeys every sign rule automatically. The built-in `abs()` gives the **absolute value**. Watch the difference between the *unary* minus that makes a number negative and the *binary* minus that subtracts – Python reads `5 − −2` as "five minus negative two".

```
print(−8 + 3)        # different signs
print(−7 * −6)       # same signs -> positive
print(5 − −2)        # subtract a negative
print(abs(−9))       # absolute value
```

```
output
−5
42
7
9
```

```
temp = −3
temp += 9            # rises 9 degrees
temp −= 5            # drops 5 degrees
print(temp)
```

```
output
1
```

Try it:

1. Predict, then run: `print(−2 * −3 * −4)`. Is the sign what the lesson predicts?
2. Use `abs()` to print the distance between `−9` and `4` on the number line (hint: `abs(−9 − 4)`).

Answer Key – Chapter 6 – Integers & Signed Numbers

6.1.Q1. -2 – it lies to the right of -9 on the number line.

6.1.Q2. 12 – absolute value is distance from zero, always non-negative.

6.1.Q3. -13 – same signs: add sizes $(8+5)$, keep negative.

6.1.Q4. -5 – different signs: $9 - 4 = 5$, sign of the larger (-9).

6.1.Q5. 13 – subtracting the opposite: $6 + 7 = 13$.

6.1.Q6. 42 – same signs multiply to a positive.

6.1.Q7. -4 – different signs give a negative quotient.

6.1.Q8. False – three (an odd number of) negative factors give a negative: -24.

6.1.Q9. $-7°$ - $(-4) + (-3) = -7$.



PART II

Factors, Multiples & Number Theory

Breaking numbers into their building blocks: factors, primes, common factors and multiples, and the distributive property that powers mental arithmetic and algebra alike.

Chapter 7 – Factors, Primes & Factor Trees · Chapter 8 – Greatest Common Factor & Least Common Multiple · Chapter 9 – The Distributive Property

Chapter 7 · Factors, Primes & Factor Trees

How to list the factors of a number, tell primes from composites, and break any whole number down into its unique prime building blocks with a factor tree.

The Idea Behind It

Every whole number above 1 has one prime fingerprint, no matter how you split it, and that is exactly why 1 is not allowed to be prime. *but why?*

Primes are the *atoms* of multiplication: numbers that cannot be broken apart. Any other number is built by multiplying atoms together. The surprise is that the recipe is **forced**, take two different routes into 36 and the bag of primes you end with is identical. Watch two factor trees race:

$$36 = 4 \times 9 = (2 \times 2) \times (3 \times 3) = 2^2 \times 3^2$$

$$36 = 6 \times 6 = (2 \times 3) \times (2 \times 3) = 2^2 \times 3^2$$

Different first cuts, **same** tips: two 2s and two 3s. This always happens, because each composite tip is itself a number that keeps splitting until only primes remain, and a prime has nowhere left to break. That guaranteed one-and-only-one answer is the **Fundamental Theorem of Arithmetic**:

$$\boxed{\text{every } n > 1 \text{ is a product of primes in exactly one way (up to order)}}$$

Now see why 1 must be banned. If 1 counted as prime, we could glue copies of it on for free and the "exactly one way" would collapse:

$$6 = 2 \times 3 = 1 \times 2 \times 3 = 1 \times 1 \times 2 \times 3 = \cdots$$

Suddenly 6 would have endlessly many factorisations, and the fingerprint would be worthless.

1 would wreck uniqueness Excluding 1 keeps every number's prime signature unique.

$$\therefore \quad 1 \text{ is neither prime nor composite}$$

So a prime is a genuine building block, and 1 stays on the sidelines precisely *so that* the one-fingerprint rule can hold.

primes are the atoms; 1 is just "multiply by nothing"

Every whole number is built from smaller whole numbers multiplied together. The number 12, for example, is 3×4, but also 2×6 and $2 \times 2 \times 3$. Understanding *which* numbers divide evenly into a given number, and which numbers cannot be broken down any further, is the foundation for fractions, for finding common denominators, and for almost everything in number theory. We start with two words that are easy to confuse.

FACTOR

A whole number that divides **exactly** into another, leaving no remainder. The factors of 12 are $1, 2, 3, 4, 6, 12$ because each of these divides 12 with remainder 0. Factors of a number are never larger than the number itself.

MULTIPLE

The result of multiplying a number by $1, 2, 3, \ldots$ – so the multiples of 12 are $12, 24, 36, 48, \ldots$ Multiples are never smaller than the number itself. *Factors divide in; multiples step out.*

Notice the mirror image: 3 is a **factor** of 12, and 12 is a **multiple** of 3. The same pair of numbers, viewed from opposite ends. Factors always come in pairs – to find them, walk upward from 1 and write each partner:

$$12 = 1 \times 12 = 2 \times 6 = 3 \times 4$$

Once the two numbers in a pair meet or cross (after 3×4 the next try, 4×3, just repeats), you have them all. Now the most important distinction in this chapter.

PRIME NUMBER

A whole number greater than 1 whose **only** factors are 1 and itself. The first primes are $2, 3, 5, 7, 11, 13, 17, 19, 23, \ldots$ Note that 2 is the only **even** prime – every other even number has 2 as an extra factor.

COMPOSITE NUMBER

A whole number greater than 1 that has **at least one** factor besides 1 and itself – that is, it can be built by multiplying smaller numbers. $12, 15, 100$ are composite. The number 1 is *neither* prime nor composite; it is a special case all on its own.

Why does 1 get excluded? Because a prime is meant to be a genuine *building block*, and allowing 1 would break the neat fact we are about to meet: that every number factors into primes in exactly one way.

The Fundamental Theorem of Arithmetic

Every whole number greater than 1 is either prime, or can be written as a product of primes in **exactly one way** (apart from the order you write them in). This unique fingerprint is called the number's **prime factorization**, and the quickest way to find it by hand is a **factor tree**.

FACTOR TREE

A diagram that repeatedly splits a number into a factor pair, branching downward, until every branch ends in a **prime**. The primes at the tips, multiplied together, are the prime factorization.

Worked examples

Example 1

List all the factors of 24.

SOLUTION

1. Walk up from 1, writing each factor and its partner: 1×24. *pairs from the ends*
2. 2×12; 3×8; 4×6.
3. Try 5: it does not divide 24, skip it. The next pair would be 6×4, which repeats – so stop. *pairs meet → done*
4. Collect both columns in order.

Factors of 24: $1, 2, 3, 4, 6, 8, 12, 24$ (eight in all).

Example 2

Is 97 prime or composite?

SOLUTION

1. A number is prime if **no** prime up to its square root divides it. Here $\sqrt{97} \approx 9.8$, so we only need to test primes $2, 3, 5, 7$. *test up to $\sqrt{n}$*
2. 97 is odd (not $\div 2$); digit sum $9+7 = 16$ is not a multiple of 3; it does not end in 0 or 5 (not $\div 5$); and $97 \div 7 \approx 13.9$ (not exact).
3. None of the primes up to $\sqrt{97}$ divide it. *nothing divides → prime*

97 **is prime.**

Example 3

Find the prime factorization of 60 with a factor tree.

SOLUTION

1. Split off any factor pair, say $60 = 6 \times 10$. *branch into a pair*
2. Neither 6 nor 10 is prime, so split again: $6 = 2 \times 3$ and $10 = 2 \times 5$. *keep splitting non-primes*
3. Now every tip – $2, 3, 2, 5$ – is prime. Multiply them: $2 \times 3 \times 2 \times 5$.
4. Gather equal primes using powers and order them.

$60 = 2^2 \times 3 \times 5$.

Example 4

Show that the starting split does not matter: factor 60 again, beginning $60 = 4 \times 15$.

SOLUTION

1. Split: $4 = 2 \times 2$ and $15 = 3 \times 5$. *different first split*
2. Tips are $2, 2, 3, 5$ – the **same multiset** as before. *same primes!*
3. This is the Fundamental Theorem in action: any route gives one answer.

$60 = 2^2 \times 3 \times 5$**, exactly as in Example 3.**

Example 5

Find the prime factorization of 360.

SOLUTION

1. $360 = 36 \times 10$. *branch into a pair*
2. $36 = 6 \times 6 = (2 \times 3) \times (2 \times 3)$, and $10 = 2 \times 5$.
3. Tips: $2, 3, 2, 3, 2, 5$ – that is three 2s, two 3s, one 5. *count each prime*

$360 = 2^3 \times 3^2 \times 5$**.**

Example 6

Are 14 and 15 relatively prime?

SOLUTION

1. Two numbers are **relatively prime** (or *coprime*) when they share **no** common factor except 1.
2. Factor each: $14 = 2 \times 7$ and $15 = 3 \times 5$. *compare the prime lists*
3. They share no prime – the factorizations have nothing in common. *no shared prime → coprime*

Yes, 14 **and** 15 **are relatively prime, even though** ***neither one is prime by itself*****.**

Example 7

List all the factors of 49, and notice the count.

SOLUTION

1. Walk up from 1 writing each pair: 1×49. *pairs from the ends*
2. Next try 7: $7 \times 7 = 49$. Here both partners are the **same**, so this pair contributes a single new factor.

square root pairs with itself

3. After 7 the partners would only repeat, so stop. *pairs meet → done*
4. Because 49 is a perfect square, it has an **odd** number of factors.

Factors of 49**:** $1, 7, 49$ **(three in all – an odd count).**

Example 8

Find the prime factorization of 198.

SOLUTION

1. Peel off the smallest prime first: 198 is even, so $198 = 2 \times 99$. *even → take a 2*
2. Now 99: its digit sum $9+9 = 18$ is a multiple of 3, so $99 = 3 \times 33$, and $33 = 3 \times 11$.
   digit-sum rule → take a 3
3. Every tip – $2, 3, 3, 11$ – is now prime (11 cannot break down). *11 is prime → stop*
4. Gather the equal primes with a power.

$198 = 2 \times 3^2 \times 11$.

Example 9

Find the highest common factor of 72 and 120 from their prime factorizations.

SOLUTION

1. Factor each: $72 = 2^3 \times 3^2$ and $120 = 2^3 \times 3 \times 5$. *factor both first*
2. The **HCF** takes each shared prime to its **lowest** power. Both share 2 and 3 (the 5 is in 120 only).
   keep only shared primes
3. Lowest powers: 2^3 (both have it) and 3^1 (the smaller of 3^2 and 3^1). *lowest power of each*
4. Multiply: $2^3 \times 3 = 8 \times 3$.

The HCF of 72 and 120 is 24.

Note: To list factors efficiently you only ever need to test divisors up to $\sqrt{n}$: every factor below the square root is paired with one above it, so finding the small one hands you the large one for free. For 36, testing up to 6 finds all six pairs.

factor multiple prime number composite number prime factorization relatively prime

⚡ Mental-Math Shortcut

Divisibility rules let you spot factors instantly – no division needed.

- **2** – the number is even (ends in $0, 2, 4, 6, 8$).
- **3** – the **digit sum** is a multiple of 3. $1+7+1 = 9$, so 171 is divisible by 3.
- **4** – the **last two digits** form a multiple of 4. $1{,}3\underline{16}$: $16 \div 4 = 4$, so yes.
- **5** – ends in 0 or 5.
- **6** – divisible by **both** 2 and 3.
- **8** – the **last three digits** form a multiple of 8.
- **9** – the **digit sum** is a multiple of 9. $5+4 = 9$, so 54 is divisible by 9.
- **10** – ends in 0.
- **11** – the **alternating digit sum** is 0 or a multiple of 11. For $2{,}915$: $2 - 9 + 1 - 5 = -11$, divisible by 11.

To start a factor tree fast, run the number past $2, 3, 5$ in your head and peel off the smallest prime you find first.

⚠ Common Mistakes

- Mixing up factors and multiples. Factors *divide into* the number and stop at it $(1..n)$; multiples *step out* from it $(n, 2n, 3n, \ldots)$. 6 is a factor of 18; 18 is a multiple of 6.
- Calling 1 prime. A prime needs **exactly two** distinct factors; 1 has only one, so it is neither prime nor composite.
- Forgetting that 2 is prime because "all primes are odd." 2 is the single even prime; the *odd* part starts at 3.
- Stopping a factor tree at a composite tip – e.g. leaving 4 or 9 unbroken. Every tip must be prime, so $4 \to 2 \times 2$ and $9 \to 3 \times 3$.
- Thinking two numbers must themselves be prime to be relatively prime. 8 and 9 are coprime although both are composite – they just share no common factor.

Quiz

7.1.Q1. List all the factors of 36.

7.1.Q2. Which of these is **prime**?

A. 51

B. 57

C. 59

D. 63

7.1.Q3. Give the prime factorization of 84.

7.1.Q4. Give the prime factorization of 200.

7.1.Q5. True or False: 1 is a prime number.

A. True

B. False

7.1.Q6. True or False: every even number greater than 2 is composite.

A. True

B. False

7.1.Q7. Use a divisibility rule: is $4{,}113$ divisible by 3?

7.1.Q8. Are 35 and 24 relatively prime?

🐍 From Concept to Code (Python)

The "leaves no remainder" test *is* the modulo operator `%`, which gives the remainder after division. `12 % 3` is `0`, so 3 divides 12. Looping a candidate divisor across `range` lets us collect every factor – and a number is prime exactly when that list has length 2.

```python
def factors(n):
    return [d for d in range(1, n + 1) if n % d == 0]

print(factors(24))
print(factors(13))          # a prime has only 1 and itself
print(len(factors(13)) == 2)
```

```
output
[1, 2, 3, 4, 6, 8, 12, 24]
[1, 13]
True
```

```python
def is_prime(n):
    if n < 2:
        return False
    d = 2
    while d * d <= n:           # only test up to sqrt(n)
        if n % d == 0:
            return False
        d += 1
    return True

print([n for n in range(2, 20) if is_prime(n)])
```

```
output
[2, 3, 5, 7, 11, 13, 17, 19]
```

Try it:

1. Write `divides(a, b)` returning `True` when `a` is a factor of `b`, then use it to print every factor of 48.
2. Modify `is_prime` to also *print* the first divisor it finds, so you can watch it reject 91 (hint: $91 = 7 \times 13$).

Answer Key – Chapter 7 – Factors, Primes & Factor Trees

7.1.Q1. $1, 2, 3, 4, 6, 9, 12, 18, 36$ – nine factors (an odd count, because 36 is a perfect square with 6×6).

7.1.Q2. 59 – no prime up to $\sqrt{59} \approx 7.7$ divides it. ($51 = 3 \times 17$, $57 = 3 \times 19$, $63 = 7 \times 9$.)

7.1.Q3. $84 = 2^2 \times 3 \times 7$ (tree: $84 = 4 \times 21 = (2 \times 2) \times (3 \times 7)$).

7.1.Q4. $200 = 2^3 \times 5^2$ (e.g. $200 = 8 \times 25$).

7.1.Q5. False – 1 is neither prime nor composite.

7.1.Q6. True – it has 2 as a factor besides 1 and itself.

7.1.Q7. Yes – digit sum $4+1+1+3 = 9$, a multiple of 3.

7.1.Q8. Yes – $35 = 5 \times 7$ and $24 = 2^3 \times 3$ share no common prime, so their only common factor is 1.

Chapter 8 · Greatest Common Factor & Least Common Multiple

How to find the greatest common factor and least common multiple of two numbers – by listing, by prime factorization, and using the shortcut that ties them together.

The Idea Behind It

The GCF and the LCM multiply back to the two numbers themselves: $\gcd(a,b) \times \operatorname{lcm}(a,b) = a \times b$**.**

but why?

Look at each prime on its own. In the factorisations of a and b a prime appears to some power in each, say p^2 in one and p^1 in the other. The **GCF** keeps the *smaller* power (the most they both share) and the **LCM** keeps the *larger* power (enough to cover either):

$$\gcd \to p^{\,\min}, \qquad \operatorname{lcm} \to p^{\,\max}$$

Here is the whole secret in one line, the rule the proof leans on, true for any two numbers:

$$\boxed{\min(x,y) + \max(x,y) = x + y}$$

The smaller and the larger together are just the *two* of them. So when you multiply GCF by LCM, that prime's exponent becomes $\min + \max = x + y$, exactly the powers in a and b added, which is what $a \times b$ carries. Do it prime by prime with $12 = 2^2 \times 3$ and $18 = 2 \times 3^2$:

$$\gcd = 2^{\,1} \times 3^{\,1} = 6, \qquad \operatorname{lcm} = 2^{\,2} \times 3^{\,2} = 36$$

$$\gcd \times \operatorname{lcm} = 2^{1+2} \times 3^{1+2} = 2^2 \times 3^2 \times (2 \times 3) = 216$$

$$\therefore\ \gcd(12,18) \times \operatorname{lcm}(12,18) = 6 \times 36 = 216 = 12 \times 18$$

So nothing is lost or doubled: the min and the max between them account for every prime factor exactly as often as $a \times b$ does. *min + max = both* That is why, once you have the GCF, the LCM is free: $\operatorname{lcm} = \dfrac{a \times b}{\gcd}$.

Two whole numbers usually share some factors and some multiples. The biggest factor they both own simplifies fractions to lowest terms; the smallest multiple they both reach is exactly the common denominator you need to add fractions. These two ideas – the **GCF** and the **LCM** – are opposite ends of the same toolkit, and they fall straight out of the prime factorizations from Chapter 7.

GREATEST COMMON FACTOR (GCF)

The largest whole number that divides **both** numbers exactly. Also called the *highest common factor* (HCF) or *greatest common divisor* (GCD). The GCF of 12 and 18 is 6, because 6 divides both and nothing larger does.

LEAST COMMON MULTIPLE (LCM)

The smallest positive number that is a multiple of **both** numbers. The LCM of 12 and 18 is 36 – the first number that appears in *both* times-tables.

Method 1 – listing

For small numbers, just list. Write out the factors (or multiples) of each, then read off the common ones.

- Factors of 12: $1, 2, 3, 4, **6, 12$. Factors of 18: $1, 2, 3, **6, 9, 18$. Greatest shared $\Rightarrow$ **GCF** $= 6$.
- Multiples of 12: $12, 24, **36, 48$. Multiples of 18: $18, **36, 54$. Least shared $\Rightarrow$ **LCM** $= 36$.

Method 2 – prime factorization

For larger numbers, factor each into primes (Chapter 7), then:

$$\mathbf{GCF} = \text{product of primes in } \mathbf{both}, \text{ each to its } \mathbf{lower} \text{ power}$$

$$\mathbf{LCM} = \text{product of } \mathbf{every} \text{ prime that appears, each to its } \mathbf{higher} \text{ power}$$

Think of it as: the GCF takes only what they *share* (the overlap), while the LCM is *greedy* and takes the most of everything.

The link between them

$$\mathbf{GCF}(a, b) \times \mathbf{LCM}(a, b) = a \times b$$

So once you have the GCF, the LCM is free: $\operatorname{lcm}(a, b) = \dfrac{a \times b}{\gcd(a, b)}$. This is usually the fastest route of all.

Worked examples

Example 1

Find the GCF and LCM of 8 and 12 by listing.

SOLUTION

1. Factors of 8: $1, 2, \boxed{4}, 8$. Factors of 12: $1, 2, 3, \boxed{4}, 6, 12$. Common: $1, 2, 4$; greatest is $\boxed{4}$. *biggest shared*
2. Multiples of 8: $8, 16, \boxed{24}, \ldots$ Multiples of 12: $12, \boxed{24}, \ldots$ First shared is $\boxed{24}$. *smallest shared*

GCF $= \boxed{4}$, **LCM** $= \boxed{24}$.

Example 2

Find the GCF of 48 and 60 using prime factorization.

SOLUTION

1. Factor: $48 = 2^4 \times 3$ and $60 = 2^2 \times 3 \times 5$.
2. Primes in **both**: 2 and 3. Take the **lower** power of each: 2^2 and 3^1. *lower wins for GCF*
3. Multiply: $2^2 \times 3 = 4 \times 3 = 12$.

GCF $= 12$.

Example 3

Find the LCM of 48 and 60.

SOLUTION

1. Same factorizations: $48 = 2^4 \times 3$, $60 = 2^2 \times 3 \times 5$.
2. Take **every** prime at its **higher** power: 2^4 (from 48), 3^1, 5^1 (from 60). *higher wins for LCM*
3. Multiply: $16 \times 3 \times 5 = 240$.

LCM $= 240$.

Example 4

Verify the product rule for 48 and 60.

SOLUTION

1. From Examples 2 and 3, GCF $= 12$ and LCM $= 240$.
2. Check $\gcd \times \operatorname{lcm}$ against $a \times b$: $12 \times 240 = 2880$. *GCF·LCM*
3. And $48 \times 60 = 2880$. They match. *a·b too*

Confirmed: $\gcd(48, 60) \times \operatorname{lcm}(48, 60) = 48 \times 60 = 2880$.

Example 5

Use the link to find the LCM of 18 and 24 quickly.

SOLUTION

1. Find the GCF first: $18 = 2 \times 3^2$, $24 = 2^3 \times 3$; shared lower powers $2^1 \times 3^1 = 6$. *GCF first*
2. Apply $\operatorname{lcm} = \dfrac{a \times b}{\gcd} = \dfrac{18 \times 24}{6}$.
3. Compute: $\dfrac{432}{6} = 72$.

LCM $= 72$ **(and GCF** $= 6$**).**

Example 6

Use the LCM to add $\frac{1}{6} + \frac{1}{8}$.

SOLUTION

1. The common denominator is $\text{lcm}(6, 8)$. Factor: $6 = 2 \times 3$, $8 = 2^3$; LCM $= 2^3 \times 3 = 24$.

   least common denom

2. Rewrite each fraction over 24: $\frac{1}{6} = \frac{4}{24}$ and $\frac{1}{8} = \frac{3}{24}$.
3. Add the numerators: $\frac{4+3}{24} = \frac{7}{24}$. *tops only*

$\frac{1}{6} + \frac{1}{8} = \frac{7}{24}$ **– this is exactly why the LCM is the *least common denominator*.**

Example 7

Find the GCF and LCM of **three** numbers: 12, 18 and 30.

SOLUTION

1. Factor every number into primes: $12 = 2^2 \times 3$, $18 = 2 \times 3^2$, $30 = 2 \times 3 \times 5$.
2. GCF takes each prime in **all three**, at its **lowest** power: 2 appears as $2^2, 2^1, 2^1$ so take 2^1; 3 appears as $3^1, 3^2, 3^1$ so take 3^1; 5 is missing from two of them, so it is out. GCF $= 2 \times 3 = 6$.

   must be in all three

3. LCM takes **every** prime that appears anywhere, at its **highest** power: 2^2 (from 12), 3^2 (from 18), 5^1 (from 30).
4. Multiply: $4 \times 9 \times 5 = 180$. *greedy: highest of each*

GCF $= 6$**, LCM** $= 180$**. (Note the product rule** $\gcd \times \text{lcm} = a \times b$ **does not extend to three numbers.)**

Example 8

Two lighthouses flash together now: one every 24 s, the other every 40 s. After how many seconds do they next flash together?

SOLUTION

1. They coincide on a time that is a multiple of **both** periods, and the *next* one is the **least** common multiple. So find $\text{lcm}(24, 40)$. *next match = LCM*
2. Factor: $24 = 2^3 \times 3$ and $40 = 2^3 \times 5$.
3. Highest power of every prime: 2^3, 3^1, 5^1.
4. Multiply: $8 \times 3 \times 5 = 120$.

They next flash together after 120 **seconds (every 2 minutes).**

Example 9

A sheet of stickers is 84 cm by 126 cm. What is the side of the **largest** identical square that tiles it with no waste, and how many squares result?

SOLUTION

1. The square's side must divide **both** 84 and 126 exactly, and the largest such side is the GCF. So find $\gcd(84, 126)$. *largest tile = GCF*
2. Factor: $84 = 2^2 \times 3 \times 7$ and $126 = 2 \times 3^2 \times 7$.
3. Shared primes at their **lower** power: 2^1, 3^1, 7^1, giving $2 \times 3 \times 7 = 42$. So each square is 42 cm on a side.
4. Count the squares: $\frac{84}{42} = 2$ across and $\frac{126}{42} = 3$ down, so $2 \times 3 = 6$ squares. *divide each side by 42*

Largest square side $= 42$ cm, giving 6 squares.

Note: When two numbers are **relatively prime** (no shared factor, Chapter 7), their GCF is 1 and their LCM is simply their product. For 9 and 10: GCF $= 1$, LCM $= 90$.

greatest common factor least common multiple common denominator lowest terms

Mental-Math Shortcut

Spotting the GCF fast. Run both numbers past the divisibility rules (Chapter 7) and peel off shared primes:

- If **both** are even, 2 is a common factor – halve them and keep going.
- If **both** digit-sums are multiples of 3, pull out a 3.
- If **both** end in 0 or 5, pull out a 5.

For 36 and 48: both even $\rightarrow$ share 2; halves $18, 24$ still both even $\rightarrow$ another 2; halves $9, 12$ have digit-sums $9, 3 \rightarrow$ share 3. Collected: $2 \times 2 \times 3 = 12 = \gcd$.

LCM trick: find the GCF first, then $\text{lcm} = \frac{a \times b}{\gcd}$ – and to avoid big products, divide *before* you multiply: $\text{lcm}(36, 48) = \frac{36}{12} \times 48 = 3 \times 48 = 144$.

⚠ Common Mistakes

- Swapping the rules: for the **GCF** take the **lower** power of shared primes; for the **LCM** take the **higher** power of every prime. Mixing them up gives a too-big GCF or too-small LCM.
- Forgetting a prime that sits in only one number when building the LCM. The LCM must contain *every* prime that appears in *either* number.
- Assuming GCF and LCM are close in size. They are usually far apart: $\gcd(48, 60) = 12$ but $\text{lcm} = 240$.
- Using the product $a \times b$ as the LCM. That only works when the GCF is 1; otherwise $\text{lcm} = \dfrac{a \times b}{\gcd}$, which is smaller.
- Reducing a fraction by a *common factor that is not the greatest* and thinking you are done – e.g. $\dfrac{24}{36} \to \dfrac{12}{18}$ still simplifies. Divide by the full GCF (12) to reach $\dfrac{2}{3}$ in one step.

📝 Quiz

8.1.Q1. Find the GCF of 24 and 36.

A. 6

B. 12

C. 72

D. 4

8.1.Q2. Find the LCM of 24 and 36.

A. 12

B. 60

C. 72

D. 864

8.1.Q3. Find the GCF of 15 and 28.

8.1.Q4. Find the LCM of 15 and 28.

8.1.Q5. Given $\gcd(a, b) = 8$ and $\text{lcm}(a, b) = 120$, what is $a \times b$?

8.1.Q6. What is the least common denominator for $\dfrac{3}{10}$ and $\dfrac{5}{12}$?

8.1.Q7. True or False: for any two numbers, the LCM is always a multiple of the GCF.

A. True

B. False

8.1.Q8. Reduce $\dfrac{42}{56}$ to lowest terms.

🐍 From Concept to Code (Python)

Python ships the GCF in the standard library as `math.gcd`. There is no built-in `lcm` before Python 3.9, but the product rule gives it in one line – and dividing first with `//` keeps the numbers small.

```
import math

print(math.gcd(48, 60))          # greatest common factor

def lcm(a, b):
    return a // math.gcd(a, b) * b   # divide before multiply

print(lcm(48, 60))
```

```
output
12
240
```

```
import math

a, b = 18, 24
g = math.gcd(a, b)
print(g * lcm(a, b) == a * b)    # verify GCF x LCM = a x b
```

```
output
True
```

Try it:

1. Use `math.gcd` to reduce the fraction `42/56`: divide both numerator and denominator by their GCF and print the result as `n/d`.
2. Write `lcm3(a, b, c)` for three numbers by chaining: `lcm(lcm(a, b), c)`. Test it on `4, 6, 10` (expect `60`).

Answer Key – Chapter 8 – Greatest Common Factor & Least Common Multiple

8.1.Q1. $12 - 24 = 2^3 \times 3$, $36 = 2^2 \times 3^2$; lower powers $2^2 \times 3 = 12$.

8.1.Q2. 72 – higher powers $2^3 \times 3^2 = 72$.

8.1.Q3. $1 - 15 = 3 \times 5$, $28 = 2^2 \times 7$ share no prime, so they are relatively prime.

8.1.Q4. 420 – coprime numbers, so LCM $= 15 \times 28 = 420$.

8.1.Q5. 960 – by the product rule $a \times b = \gcd \times \operatorname{lcm} = 8 \times 120 = 960$.

8.1.Q6. 60 – $\operatorname{lcm}(10, 12)$: $10 = 2 \times 5$, $12 = 2^2 \times 3$, so $2^2 \times 3 \times 5 = 60$.

8.1.Q7. True – the GCF divides both numbers, so it divides their LCM as well.

8.1.Q8. $\frac{3}{4}$ – $\gcd(42, 56) = 14$, and $42 \div 14 = 3$, $56 \div 14 = 4$.

Chapter 9 · The Distributive Property

How to multiply across a sum with the distributive property $a(b+c) = ab + ac$ – to do mental arithmetic, expand expressions, and factor a common term back out.

The Idea Behind It

Multiplying across a bracket hits every term inside: $a(b+c) = ab + ac$. but why?

A product like $a \times (\text{something})$ is the **area** of a rectangle a tall and "something" wide. So picture a rectangle that is a tall and $(b+c)$ wide. Its area, all at once, is the law you already trust, area equals height times width:

$$\boxed{\text{area} = \text{height} \times \text{width}} \quad \Longrightarrow \quad \text{area} = a\,(b+c)$$

Now slice that same rectangle with one vertical cut at width b. You have not added or removed any space, just split it into a b-wide piece and a c-wide piece sitting side by side:

$$a\,(b+c) = \underbrace{a\,b}_{\text{left piece}} + \underbrace{a\,c}_{\text{right piece}}$$

The whole must equal the sum of its parts, so the two areas add back to the original. Put real numbers on it, a 7-tall strip that is $4+3$ wide:

$$7\,(4+3) = 7 \times 4 + 7 \times 3 = 28 + 21 = 49$$

$$\therefore\ a(b+c) = ab + ac$$

The factor reaches *both* terms because it is the height of *both* pieces.

cut the rectangle, the area cannot change That single picture is the engine behind mental multiplication, expanding brackets, and factoring.

Suppose you buy 7 gift bags, each holding a toy costing 4 pounds and a pen costing 3 pounds. You can total the spend two ways: work out one bag ($4+3=7$ pounds) then multiply by 7, **or** total all the toys ($7 \times 4 = 28$) and all the pens ($7 \times 3 = 21$) and add. Both give 49 pounds. That equality is the **distributive property**, and it is the single most useful rule in arithmetic and algebra – it powers mental multiplication, expanding brackets, and factoring.

THE DISTRIBUTIVE PROPERTY

Multiplying a number by a sum equals multiplying it by each part and adding:

$$a(b+c) = ab + ac$$

The outside factor is "**distributed**" to every term inside the bracket. It works for subtraction too – just carry the sign:

$$a(b-c) = ab - ac$$

Read **left to right**, $a(b+c) \to ab + ac$ is called **expanding** (removing the bracket). Read **right to left**, $ab + ac \to a(b+c)$ is called **factoring** (pulling a common factor out front). They are the same rule run in opposite directions.

Why it works

A rectangle a tall and $(b + c)$ wide can be sliced into an $a \times b$ piece and an $a \times c$ piece. The whole area equals the sum of the parts: $a(b + c) = ab + ac$. The picture makes the algebra obvious.

Worked examples

Example 1

Expand $6(10 + 3)$ and check.

SOLUTION

1. Distribute the 6 to each term: $6 \times 10 + 6 \times 3$. *6 hits both*
2. Compute each product: $60 + 18 = 78$.
3. Check directly: the bracket is $10 + 3 = 13$, and $6 \times 13 = 78$. Match. *same answer*

$6(10 + 3) = 60 + 18 = 78$.

Example 2

Use distribution to compute 7×23 in your head.

SOLUTION

1. Break 23 into a friendly sum: $23 = 20 + 3$. *split to round*
2. Distribute: $7 \times 20 + 7 \times 3$.
3. Add the easy products: $140 + 21 = 161$.

$7 \times 23 = 161$.

Example 3

Expand $4(x + 5)$.

SOLUTION

1. The factor 4 multiplies **both** terms inside the bracket. *reaches both*
2. $4 \times x = 4x$ and $4 \times 5 = 20$.
3. A letter times a number just gets written side by side as a **coefficient**.

$4(x + 5) = 4x + 20$.

Example 4

Expand $-3(2x - 5)$, watching the signs.

SOLUTION

1. Distribute -3 to each term, keeping the sign attached: $(-3)(2x) + (-3)(-5)$.
2. First product: $-3 \times 2x = -6x$. Second: a negative times a negative is positive, $(-3)(-5) = +15$.

   − × − = +
3. Combine.

$-3(2x - 5) = -6x + 15$.

Example 5

Factor $12x + 18$ by pulling out the greatest common factor.

SOLUTION

1. Find the GCF of the coefficients (Chapter 8): $\gcd(12, 18) = 6$. *pull this out*
2. Write each term as 6 times something: $12x = 6 \cdot 2x$ and $18 = 6 \cdot 3$.
3. Place the common 6 outside a bracket: $6(2x + 3)$.
4. Check by expanding: $6 \times 2x + 6 \times 3 = 12x + 18$. Match. *reverse gear*

$12x + 18 = 6(2x + 3)$.

Example 6

Expand and simplify $5(2a + 3) + 2(a - 4)$.

SOLUTION

1. Distribute each bracket separately: $5 \cdot 2a + 5 \cdot 3 = 10a + 15$, and $2 \cdot a + 2 \cdot (-4) = 2a - 8$.

   expand each
2. Write the full expansion: $10a + 15 + 2a - 8$.
3. Collect like terms: the a-terms $10a + 2a = 12a$; the numbers $15 - 8 = 7$. *group alikes*

$5(2a + 3) + 2(a - 4) = 12a + 7$.

Example 7

Two-digit multiplication as double distribution: 23×14.

SOLUTION

1. Split both numbers: $23 = 20 + 3$, $14 = 10 + 4$, so $23 \times 14 = (20 + 3)(10 + 4)$. *each by each*
2. Distribute every term against every term: $20 \times 10 + 20 \times 4 + 3 \times 10 + 3 \times 4$.
3. Compute: $200 + 80 + 30 + 12$.
4. Add: 322.

$23 \times 14 = 322$ – **this "each-by-each" rule is exactly the FOIL method you will meet in algebra.**

Example 8

Distribute a **variable** factor: expand $2x(3x + 5)$.

SOLUTION

1. The whole factor $2x$ multiplies **each** term inside the bracket. *2x hits both*
2. First term: $2x \times 3x = 6x^2$ – multiply the numbers ($2 \times 3 = 6$) and the letters ($x \times x = x^2$).
3. Second term: $2x \times 5 = 10x$.
4. Add the two products.

$2x(3x + 5) = 6x^2 + 10x$.

Example 9

Multiply two brackets (FOIL in symbols): expand $(x + 4)(x + 3)$.

SOLUTION

1. Treat the first bracket as the outside factor and distribute it: $(x + 4) \cdot x + (x + 4) \cdot 3$.

   split the 2nd bracket
2. Distribute again inside each piece: $x \cdot x + 4 \cdot x = x^2 + 4x$, and $x \cdot 3 + 4 \cdot 3 = 3x + 12$.
3. Write the full expansion: $x^2 + 4x + 3x + 12$.
4. Collect the like middle terms: $4x + 3x = 7x$. *add the x-terms*

$(x + 4)(x + 3) = x^2 + 7x + 12$ – **every term of the first bracket times every term of the second.**

Example 10

Factor out a **variable** common factor: write $6x^2 + 9x$ as a product.

SOLUTION

1. Look at the numbers and the letters separately. Number GCF: $\gcd(6, 9) = 3$. Letter part: both terms contain at least one x, so x comes out too. The common factor is $3x$. *pull out 3x*
2. Write each term as $3x$ times what is left: $6x^2 = 3x \cdot 2x$ and $9x = 3x \cdot 3$.
3. Place the common $3x$ outside a bracket: $3x(2x + 3)$.
4. Check by expanding: $3x \times 2x + 3x \times 3 = 6x^2 + 9x$. Match. *reverse gear*

$6x^2 + 9x = 3x(2x + 3)$.

Note: Distribution applies only across $+$ and $-$ *inside the bracket*. It does **not** reach across a multiplication or a power: $a(bc) \neq ab \cdot ac$, and $(a + b)^2 \neq a^2 + b^2$. Only sums and differences get distributed over.

distributive property expand factor (verb) coefficient like terms

⚡ Mental-Math Shortcut

Break-apart products. Distribution is the engine behind nearly every mental-multiplication trick: split one factor into round-number pieces, multiply each, then add.

- **By breakdown** – split into a sum: $7 \times 23 = 7(20 + 3) = 140 + 21 = 161$.
- **By complement** – split into a difference when a factor is just under a round number: $7 \times 49 = 7(50 - 1) = 350 - 7 = 343$.
- **Doubling spread out**: $8 \times 26 = 8(25 + 1) = 200 + 8 = 208$.

Always break the factor whose pieces land on round numbers. For 6×198 use $6(200 - 2) = 1200 - 12 = 1188$, not $6(190 + 8)$.

⚠ Common Mistakes

- Distributing to only the first term: $3(x + 4) = 3x + 4$. Wrong – the 3 hits **every** term: $3x + 12$.
- Dropping the sign on a negative factor: $-2(x - 5) = -2x - 10$. Wrong – $(-2)(-5) = +10$, so it is $-2x + 10$.
- Trying to distribute over multiplication or a power: $(a + b)^2 = a^2 + b^2$. Wrong – $(a + b)^2 = a^2 + 2ab + b^2$; the property only spreads over $+$ and $-$.
- Factoring out less than the full GCF: $12x + 18 = 2(6x + 9)$ is *incompletely* factored. Pull out the GCF 6 to get $6(2x + 3)$.
- Combining unlike terms after expanding: $4x + 20$ is **not** $24x$. A coefficient term and a constant cannot merge.

Quiz

9.1.Q1. Expand $5(x + 7)$.

A. $5x + 7$

B. $5x + 35$

C. $x + 35$

D. $12x$

9.1.Q2. Expand $8(2x - 3)$.

9.1.Q3. Expand $-4(3 - y)$.

A. $-12 - 4y$

B. $-12 + 4y$

C. $12 + 4y$

D. $12 - 4y$

9.1.Q4. Use distribution to compute 9×34 mentally.

9.1.Q5. Factor $20x + 30$ completely.

9.1.Q6. Simplify $3(x + 2) + 4(x + 1)$.

9.1.Q7. True or False: $a(b + c) = ab + ac$ for all numbers a, b, c.

A. True

B. False

9.1.Q8. True or False: $2(x \times y) = 2x \times 2y$.

A. True

B. False

🐍 From Concept to Code (Python)

A Python **function** packages a calculation so you can reuse it and `return` the result. We can write both sides of the distributive law as functions and let Python confirm they always agree – `==` returns a Boolean `True`/`False` (Chapter 2).

```
def left(a, b, c):
    return a * (b + c)       # the bracket form

def right(a, b, c):
    return a * b + a * c     # the expanded form

print(left(7, 20, 3), right(7, 20, 3))
print(left(7, 20, 3) == right(7, 20, 3))
```

```
output
161 161
True
```

```
# the identity holds for negatives too
for a, b, c in [(3, 5, 2), (-4, 2, -5), (6, 200, -2)]:
    print(a * (b + c) == a * b + a * c)
```

```
output
True
True
True
```

Try it:

1. Write `expand(a, b, c)` that *prints* the worked line, e.g. `7*(20+3) = 140 + 21 = 161`, using an f-string.
2. Check that distribution does **not** hold for powers: print whether `(a+b)**2 == a**2 + b**2` for `a, b = 3, 4`. Why is it `False`?

Answer Key – Chapter 9 – The Distributive Property

9.1.Q1. $5x + 35$ – distribute the 5 to both terms.

9.1.Q2. $16x - 24$ – $8 \times 2x = 16x$ and $8 \times (-3) = -24$.

9.1.Q3. $-12 + 4y$ – $(-4)(3) = -12$ and $(-4)(-y) = +4y$.

9.1.Q4. 306 – $9(30 + 4) = 270 + 36 = 306$.

9.1.Q5. $10(2x + 3)$ – the GCF of 20 and 30 is 10.

9.1.Q6. $7x + 10$ – expand to $3x + 6 + 4x + 4$, then collect like terms.

9.1.Q7. True – this is the distributive property, valid for every number.

9.1.Q8. False – distribution applies over $+$ and $-$, not multiplication; $2(xy) = 2xy$, while $2x \times 2y = 4xy$.



PART III

Exponents, Roots & Scientific Notation

Repeated multiplication and its inverse: the exponent laws, square roots, powers of ten, scientific notation, and the unit-fraction method for converting between units.

Chapter 10 – Exponents & the Laws of Powers · Chapter 11 – Square Roots · Chapter 12 – Scientific Notation · Chapter 13 – Unit Conversions

Chapter 10 · Exponents & the Laws of Powers

How to read powers, and how to multiply, divide and nest them using the laws of exponents – including the zero and negative exponents, and the special powers of 10.

The Idea Behind It

A negative power is a fraction: $a^{-1} = \frac{1}{a}$, and in general $a^{-n} = \frac{1}{a^n}$. *but why?*

An exponent counts copies of the base, so stepping it **up** by 1 multiplies by a, and stepping it **down** by 1 must **divide** by a. A negative exponent is simply what you reach by stepping below zero, and it falls straight out of the **quotient law**:

$$\boxed{\frac{a^m}{a^n} = a^{m-n}}$$

Read the -1 as $0-1$ and run that law in reverse, using the two facts $a^0 = 1$ and $a^1 = a$:

$$a^{-1} = a^{0-1} = \frac{a^0}{a^1} = \frac{1}{a^1} = \frac{1}{a}$$

$$\therefore\ a^{-1} = \frac{1}{a}, \qquad a^{-n} = \frac{1}{a^n}$$

The minus sign in the exponent never makes the number negative; it just **flips** it underneath a 1.

divide, don't negate

Writing $2 \times 2 \times 2 \times 2 \times 2$ over and over is tiring, and easy to miscount. **Exponents** are a shorthand for repeated multiplication: 2^5 means "multiply five 2s together", which equals 32. The same idea lets us write enormous and tiny numbers compactly – a computer with 2^{32} addresses, or a virus 10^{-7} metres across – and a small set of *laws* lets us combine such powers without ever expanding them.

POWER, BASE AND EXPONENT

In the power a^n, the **base** a is the number being multiplied and the **exponent** (or **index**) n counts how many copies are multiplied:

$$a^n = \underbrace{a \times a \times \cdots \times a}_{n \text{ factors}}$$

We read a^2 as "a squared", a^3 as "a cubed", and a^n as "a to the nth".

The exponent only attaches to whatever is *directly* beneath it. So $-3^2 = -(3 \times 3) = -9$, whereas $(-3)^2 = (-3) \times (-3) = 9$. The brackets change the base from 3 to -3.

The laws of powers

Every law below can be discovered just by writing the factors out. They hold for any base and are the heart of this chapter.

1. Product law – *same base, multiply ⇒ add the exponents.* Counting factors, $a^m \times a^n$ has m then another n copies of a:

$$a^m \times a^n = a^{m+n}$$

2. Quotient law – *same base, divide ⇒ subtract the exponents.* Each factor on the bottom cancels one on the top:

$$\frac{a^m}{a^n} = a^{m-n} \qquad (a \neq 0)$$

3. Power of a power – *raise a power to a power ⇒ multiply the exponents.* You take n copies of a^m:

$$(a^m)^n = a^{m \cdot n}$$

4. Power of a product (and quotient) – *the exponent spreads onto each factor:*

$$(ab)^n = a^n\, b^n \qquad \left(\frac{a}{b}\right)^n = \frac{a^n}{b^n}$$

5. Zero exponent – *anything (except 0) to the power 0 is 1.* This is forced by the quotient law: $\frac{a^n}{a^n} = a^{n-n} = a^0$, and that fraction is plainly 1:

$$a^0 = 1 \qquad (a \neq 0)$$

6. Negative exponent – *a negative index means reciprocal.* Again from the quotient law, $\frac{a^0}{a^n} = a^{0-n} = a^{-n}$, and the left side is $\frac{1}{a^n}$:

$$a^{-n} = \frac{1}{a^n} \qquad (a \neq 0)$$

Worked examples

Example 1

Product law. Simplify each: (a) $2^3 \times 2^4$, (b) $5^6 \times 5$, (c) $x^2 \times x^5 \times x$.

SOLUTION

1. (a) Same base 2, so add exponents: $2^{3+4} = 2^7 = 128$. same base ⇒ add
2. (b) A lone 5 is 5^1: $5^{6+1} = 5^7$.
3. (c) Add all three: $x^{2+5+1} = x^8$.

(a) $2^7 = 128$ **(b)** 5^7 **(c)** x^8

Example 2

Quotient law. Simplify each: (a) $\frac{7^8}{7^3}$, (b) $\frac{10^5}{10^5}$, (c) $\frac{y^9}{y^2}$.

SOLUTION

1. (a) Same base 7, subtract exponents: $7^{8-3} = 7^5$. *divide ⇒ subtract*
2. (b) $10^{5-5} = 10^0 = 1$ – the top and bottom are identical.
3. (c) $y^{9-2} = y^7$.

(a) 7^5 **(b)** 1 **(c)** y^7

Example 3

Power of a power. Simplify each: (a) $(3^2)^4$, (b) $(x^5)^3$, (c) $(2^3)^2 \times 2$.

SOLUTION

1. (a) Multiply exponents: $3^{2\times 4} = 3^8 = 6561$. *power of power ⇒ ×*
2. (b) $x^{5\times 3} = x^{15}$.
3. (c) Inner power first: $(2^3)^2 = 2^6$; then the product law with 2^1: $2^{6+1} = 2^7 = 128$.

(a) $3^8 = 6561$ **(b)** x^{15} **(c)** $2^7 = 128$

Example 4

Power of a product / quotient. Simplify each: (a) $(2x)^3$, (b) $(5y^2)^2$, (c) $\left(\frac{a}{3}\right)^2$.

SOLUTION

1. (a) The exponent lands on **both** factors: $2^3 x^3 = 8x^3$. *hits every factor*
2. (b) $5^2 (y^2)^2 = 25y^4$ – note the power-of-a-power on y.
3. (c) Top and bottom each get squared: $\frac{a^2}{3^2} = \frac{a^2}{9}$.

(a) $8x^3$ **(b)** $25y^4$ **(c)** $\frac{a^2}{9}$

Example 5

Zero exponent. Evaluate each: (a) 9^0, (b) $(-4)^0$, (c) $3x^0$, (d) $(3x)^0$.

SOLUTION

1. (a) Any nonzero base to the 0 is 1: $9^0 = 1$. ^0 ⇒ 1
2. (b) The base is -4 (in brackets), still nonzero: $(-4)^0 = 1$.
3. (c) The exponent sits on x only, so $x^0 = 1$ and $3x^0 = 3 \times 1 = 3$. only x is raised
4. (d) Now the whole $3x$ is the base: $(3x)^0 = 1$.

(a) 1 **(b)** 1 **(c)** 3 **(d)** 1

Example 6

Negative exponent. Write each without a negative exponent: (a) 2^{-3}, (b) $5x^{-2}$, (c) $\dfrac{1}{4^{-2}}$.

SOLUTION

1. (a) Reciprocal of 2^3: $2^{-3} = \dfrac{1}{2^3} = \dfrac{1}{8}$. −n ⇒ flip
2. (b) The negative index is on x alone: $5x^{-2} = \dfrac{5}{x^2}$.
3. (c) A negative exponent in the **denominator** flips up: $\dfrac{1}{4^{-2}} = 4^2 = 16$.

(a) $\frac{1}{8}$ **(b)** $\dfrac{5}{x^2}$ **(c)** 16

Example 7

Powers of 10 (mixed laws). Evaluate or simplify: (a) 10^3, (b) $10^4 \times 10^2$, (c) $\dfrac{10^6}{10^9}$, (d) $(10^2)^3$, (e) 10^{-3}.

SOLUTION

1. (a) $10^3 = 1000$ – the exponent is just the number of zeros after the 1. count the zeros
2. (b) Product law: $10^{4+2} = 10^6 = 1{,}000{,}000$.
3. (c) Quotient law: $10^{6-9} = 10^{-3} = \dfrac{1}{1000} = 0.001$.
4. (d) Power of a power: $10^{2 \times 3} = 10^6$.
5. (e) Negative power of 10: $10^{-3} = 0.001$ (three places after the point).

(a) 1000 **(b)** 10^6 **(c)** $10^{-3} = 0.001$ **(d)** 10^6 **(e)** 0.001

Example 8

Two laws together. Simplify $\dfrac{2^5 \times 2^4}{2^3}$.

SOLUTION

1. Top first – product law (same base, **add**): $2^{5+4} = 2^9$, so the expression is $\dfrac{2^9}{2^3}$. *multiply ⇒ add*
2. Now the quotient law (same base, **subtract**): $2^{9-3} = 2^6$. *divide ⇒ subtract*
3. Evaluate: $2^6 = 64$.

$2^6 = 64$

Example 9

Quotient giving a negative exponent. Simplify $\dfrac{(x^2)^3}{x^8}$ and write it without a negative exponent.

SOLUTION

1. Top – power of a power (**multiply**): $(x^2)^3 = x^6$, giving $\dfrac{x^6}{x^8}$. *power of power ⇒ ×*
2. Quotient law: $x^{6-8} = x^{-2}$ – the exponent goes negative because the bottom has more factors.
   6 − 8 = −2
3. A negative exponent means reciprocal: $x^{-2} = \dfrac{1}{x^2}$. *−n ⇒ flip*

$\dfrac{1}{x^2}$

Example 10

Everything at once. Simplify $\dfrac{(2a^3b)^2 \cdot a^4}{a^2b}$.

SOLUTION

1. Expand the power-of-a-product on top – the exponent 2 lands on **every** factor: $(2a^3b)^2 = 2^2(a^3)^2b^2 = 4a^6b^2$. *hits each factor*
2. Multiply that by the a^4 also on top (product law on the as): $4a^6b^2 \cdot a^4 = 4a^{10}b^2$, so the expression is $\dfrac{4a^{10}b^2}{a^2b}$.
3. Now the quotient law on each letter separately: $a^{10-2} = a^8$ and $b^{2-1} = b^1$. *subtract per base*
4. The 4 has nothing to cancel, so collect: $4a^8b$.

$4a^8b$

Note: A positive power of 10 counts the zeros: 10^n is 1 followed by n zeros. A negative power of 10 counts the decimal places: 10^{-n} is the decimal point followed by $(n-1)$ zeros and a 1. This is exactly why scientific notation (next chapters) leans on powers of 10.

Note: The laws only combine powers of the **same base**. $2^3 \times 5^2$ cannot be merged into a single power – you must evaluate $8 \times 25 = 200$. And $a^m + a^n$ does **not** simplify by any law; addition is not multiplication.

⚡ Mental-Math Shortcut

To square a number that **ends in 5**, drop the 5, multiply the remaining "tens" part n by the next whole number $n+1$, and write **25** on the end. That works because $(10n+5)^2 = 100\,n(n+1) + 25$.

- 25^2: here $n = 2$, so $n(n+1) = 2 \times 3 = 6 \Rightarrow$ **625**.
- 35^2: $n = 3$, $3 \times 4 = 12 \Rightarrow$ **1225**.
- 85^2: $n = 8$, $8 \times 9 = 72 \Rightarrow$ **7225**.

For **powers of 2**, memorise the first few – $2, 4, 8, 16, 32, 64, 128, 256, 512, 1024$ – then reach any higher one by the product law: $2^{10} = 1024$, so $2^{12} = 2^{10} \times 2^2 = 1024 \times 4 = 4096$.

⚠ Common Mistakes

- Adding the bases instead of keeping them: $2^3 \times 2^4 = 2^7$, **not** 4^7. The base stays 2; only the exponents add.
- Multiplying exponents in the product law: $a^m \times a^n = a^{m+n}$ (add), while $(a^m)^n = a^{mn}$ (multiply). Mixing these up is the most common slip.
- Forgetting the exponent spreads onto every factor: $(2x)^3 = 8x^3$, not $2x^3$. Likewise $(5y^2)^2 = 25y^4$, not $5y^4$.
- Thinking $a^0 = 0$. Any nonzero base to the power 0 equals 1.
- Reading -3^2 as 9. Without brackets the power binds first: $-3^2 = -(3^2) = -9$; only $(-3)^2 = 9$.

Quiz

10.1.Q1. Simplify $3^4 \times 3^2$.

A. 3^8

B. 3^6

C. 9^6

D. 3^2

10.1.Q2. Simplify $\frac{x^7}{x^3}$.

A. x^4

B. x^{10}

C. x^{21}

D. x^2

10.1.Q3. Simplify $(2^3)^2$.

A. 2^5

B. 2^9

C. 2^6

D. 2^1

10.1.Q4. Expand $(3x)^2$.

10.1.Q5. Evaluate 7^0.

A. 0

B. 1

C. 7

D. undefined

10.1.Q6. Write 5^{-2} as a fraction.

10.1.Q7. Evaluate 10^4.

10.1.Q8. Simplify $\frac{10^5}{10^8}$ as a power of 10.

10.1.Q9. True or False: $2^3 + 2^3 = 2^6$.

A. True

B. False

10.1.Q10. Simplify $(2a^2)^3$.

🐍 From Concept to Code (Python)

Python writes powers with the `**` operator, or with the built-in `pow(base, exp)`. Both follow the same laws you just learned – Python is simply doing the multiplication for you. Negative and zero exponents work too; a negative exponent returns a `float`.

```
print(2 ** 5)        # 2 to the 5th
print(pow(3, 4))     # same as 3 ** 4
print(7 ** 0)        # zero exponent
print(2 ** -3)       # negative exponent -> reciprocal
print(10 ** 6)       # a power of ten
```

```
output
32
81
1
0.125
1000000
```

```
# the product law in action: a**m * a**n == a**(m+n)
print(2**3 * 2**4 == 2**(3+4))
print((2**3) ** 2 == 2 ** (3*2))   # power of a power
```

```
output
True
True
```

Try it:

1. Predict, then run: `print(-3 ** 2)` and `print((-3) ** 2)`. Why do they differ?
2. Use `**` to print the first ten powers of 2 (from `2**1` to `2**10`) with a `for` loop over `range(1, 11)`.

Answer Key – Chapter 10 – Exponents & the Laws of Powers

10.1.Q1. 3^6 – same base, add exponents $4 + 2$.

10.1.Q2. x^4 – same base, subtract exponents $7 - 3$.

10.1.Q3. $2^6 = 64$ – power of a power, multiply 3×2.

10.1.Q4. $9x^2$ – the exponent lands on both factors: 3^2x^2.

10.1.Q5. 1 – any nonzero base to the power 0 is 1.

10.1.Q6. $\frac{1}{25}$ – a negative exponent means the reciprocal: $\frac{1}{5^2}$.

10.1.Q7. 10000 – 1 followed by four zeros.

10.1.Q8. 10^{-3} (i.e. 0.001) – subtract exponents $5 - 8$.

10.1.Q9. False – that is $8 + 8 = 16 = 2^4$; the exponent laws are for multiplying, not adding.

10.1.Q10. $8a^6$ – cube each factor: $2^3(a^2)^3$.

Chapter 11 · Square Roots

How to recognise perfect squares, find and estimate square roots, simplify roots like $\sqrt{50}$, and see why squaring and square-rooting undo each other.

The Idea Behind It

$\sqrt{2}$ can never be written as an exact fraction, no decimal and no $\frac{p}{q}$ ever lands on it. *but why?*

This one we prove by **contradiction**: assume the opposite and watch it self-destruct. The one fact we lean on is about **parity**, odd versus even:

$$\boxed{\text{if } p^2 \text{ is even, then } p \text{ is even}}$$

(An odd number times itself is odd, so an even square can only have come from an even root.) Now suppose $\sqrt{2}$ *did* equal a fraction, written in **lowest terms** so p and q share no common factor. Square both sides and clear the denominator:

$$\sqrt{2} = \frac{p}{q} \implies 2 = \frac{p^2}{q^2} \implies p^2 = 2q^2$$

The right side is twice something, so p^2 is even, and by the boxed fact p is even. Write $p = 2k$ and substitute back:

$$(2k)^2 = 2q^2 \implies 4k^2 = 2q^2 \implies q^2 = 2k^2$$

Now q^2 is even too, so q is **also** even. But if p and q are both even they share the factor 2, flatly contradicting "lowest terms". The assumption breaks:

$$\therefore\ \sqrt{2} \text{ is irrational} \quad (\text{no fraction equals it})$$

So $\sqrt{2} = 1.41421356\ldots$ runs on forever with no repeating pattern, a genuinely new kind of number living between the fractions.

assume it is a fraction, force a contradiction, conclude it cannot be

Squaring a number means multiplying it by itself: $7^2 = 49$. The **square root** asks the reverse question – *what number, when squared, gives this?* Since $7 \times 7 = 49$, the square root of 49 is 7. Roots appear everywhere a squared quantity needs undoing: the side of a square of known area, the distance across a grid, the standard deviation of a data set.

SQUARE ROOT

The **square root** of a non-negative number a, written $\sqrt{a}$, is the non-negative number whose square is a:

$$\sqrt{a} = b \quad \text{means} \quad b \geq 0 \ \text{ and } \ b^2 = a.$$

The sign $\sqrt{\ }$ is the **radical**; the number underneath is the **radicand**.

A perfect square has a whole-number root. It pays to know them by heart:

n	1	2	3	4	5	6	7	8	9	10	11	12
n^2	1	4	9	16	25	36	49	64	81	100	121	144

Squaring and taking a square root are **inverse operations** – each undoes the other (for non-negative numbers): $\sqrt{n^2} = n$ and $(\sqrt{n})^2 = n$. Note that a positive number technically has two numbers that square to it ($5^2 = 25$ and $(-5)^2 = 25$), but the radical symbol always means the non-negative one, called the **principal** root. Also, negative numbers have no real square root, because no real number squared is negative.

Estimating a root that is not perfect

Most roots are irrational decimals. To estimate $\sqrt{N}$, trap N between the two nearest perfect squares; the root lies between their roots, closer to whichever square N is nearer.

Simplifying a square root

A root simplifies when the radicand hides a **perfect-square factor**. Pull that factor out using $\sqrt{a\,b} = \sqrt{a} \times \sqrt{b}$:

$$\sqrt{a\,b} = \sqrt{a}\,\sqrt{b}, \qquad \sqrt{\frac{a}{b}} = \frac{\sqrt{a}}{\sqrt{b}}$$

Worked examples

Example 1

Find (a) $\sqrt{64}$, (b) $\sqrt{144}$, (c) $\sqrt{0}$ and explain why $\sqrt{-9}$ has no real value.

SOLUTION

1. (a) $8 \times 8 = 64$, so $\sqrt{64} = 8$. *what squares to it?*
2. (b) $12 \times 12 = 144$, so $\sqrt{144} = 12$.
3. (c) $0 \times 0 = 0$, so $\sqrt{0} = 0$.
4. Any real number squared is ≥ 0, so nothing squares to -9: $\sqrt{-9}$ is not a real number. *no real root*

(a) 8 (b) 12 (c) 0 ; $\sqrt{-9}$ undefined in the reals

Example 2

Estimate $\sqrt{50}$ to one decimal place without a calculator.

SOLUTION

1. Trap it between perfect squares: $49 < 50 < 64$, so $7 < \sqrt{50} < 8$. *trap it*
2. 50 is just above $49 = 7^2$, so the root is a hair above 7.
3. Test $7.1^2 = 50.41$ – slightly too big; $7.07^2 \approx 49.98$ – almost exact.

$\sqrt{50} \approx 7.1$

Example 3

Estimate $\sqrt{30}$ and state which whole number it is closest to.

SOLUTION

1. $25 < 30 < 36$, so $5 < \sqrt{30} < 6$. *between 5 and 6*
2. 30 is roughly midway but a touch nearer 25, so the root is around 5.5.
3. Check: $5.5^2 = 30.25$ – very close.

$\sqrt{30} \approx 5.5$ **(nearest whole number 5)**

Example 4

Simplify $\sqrt{72}$.

SOLUTION

1. Find the largest perfect-square factor of 72: $72 = 36 \times 2$ and 36 is a perfect square. *spot the square*
2. Split the root: $\sqrt{72} = \sqrt{36} \times \sqrt{2}$.
3. Take the root of 36: $\sqrt{36} = 6$, leaving $6\sqrt{2}$.

$\sqrt{72} = 6\sqrt{2}$

Example 5

Simplify $\sqrt{200} + \sqrt{18}$.

SOLUTION

1. $\sqrt{200} = \sqrt{100 \times 2} = 10\sqrt{2}$.
2. $\sqrt{18} = \sqrt{9 \times 2} = 3\sqrt{2}$.
3. Both are now multiples of $\sqrt{2}$, so add the coefficients: $10\sqrt{2} + 3\sqrt{2} = 13\sqrt{2}$. *like roots add*

$13\sqrt{2}$

Example 6

A square patio has area $225\ \text{m}^2$. How long is each side?

SOLUTION

1. Area of a square is side^2, so $\text{side} = \sqrt{225}$. *undo the square*
2. $15 \times 15 = 225$, so $\sqrt{225} = 15$.

Each side is 15 m.

Example 7

Briefly: cube roots. Find (a) $\sqrt[3]{27}$ and (b) $\sqrt[3]{125}$.

SOLUTION

1. A **cube root** undoes cubing: $\sqrt[3]{a} = b$ means $b^3 = a$. *what cubes to it?*
2. (a) 3^3 $= 27$, so $\sqrt[3]{27} =$ 3 .
3. (b) 5^3 $= 125$, so $\sqrt[3]{125} =$ 5 . (Unlike square roots, cube roots of negatives *do* exist: $\sqrt[3]{-8} = -2$.)

(a) 3 **(b)** 5

Example 8

Simplify the root of a fraction: $\sqrt{\frac{48}{25}}$.

SOLUTION

1. Split the root over top and bottom (the quotient rule): $\sqrt{\frac{48}{25}} = \frac{\sqrt{48}}{\sqrt{25}}$. *root top and bottom*
2. Bottom is a perfect square: $\sqrt{25} =$ 5 .
3. Top hides a perfect-square factor: $48 =$ 16 $\times 3$, so $\sqrt{48} = \sqrt{16}\,\sqrt{3} =$ $4\sqrt{3}$. *pull out the 16*
4. Combine: $\frac{4\sqrt{3}}{5} =$ $\frac{4\sqrt{3}}{5}$.

$\sqrt{\frac{48}{25}} = \frac{4\sqrt{3}}{5}$

Example 9

Multiply then simplify: $\sqrt{6} \times \sqrt{15}$.

SOLUTION

1. Combine under one radical with $\sqrt{a}\,\sqrt{b} = \sqrt{ab}$: $\sqrt{6} \times \sqrt{15} =$ $\sqrt{90}$. *multiply under the root*
2. Find the largest perfect-square factor of 90: $90 =$ 9 $\times 10$, and 9 is a perfect square.
3. Split and take the root of 9: $\sqrt{90} = \sqrt{9}\,\sqrt{10} =$ 3 $\sqrt{10}$. *9 is a square*

$\sqrt{6} \times \sqrt{15} = 3\sqrt{10}$

Example 10

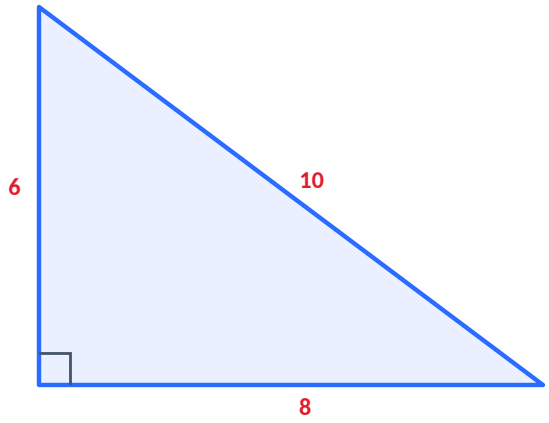
A right-angled triangle has legs 6 and 8. Use Pythagoras to find the hypotenuse c.

SOLUTION

1. Pythagoras says $c^2 = a^2 + b^2$ for the two legs a, b. Substitute: $c^2 = 6^2 + 8^2$. *legs squared, added*
2. Work out the squares and add: $36 + 64 = 100$, so $c^2 = 100$.
3. Undo the square with a root: $c = \sqrt{100} = 10$ (take the positive root – a length cannot be negative).

undo the square

$c = 10$.



The right-angled triangle of Example 10: legs 6 and 8 meet at the right angle, and the hypotenuse is $\sqrt{6^{2}+8^{2}}=\sqrt{100}=10$.

Note: Treat $\sqrt{}$ like a bracket: finish everything under the radical before combining it with the outside. So $\sqrt{9+16} = \sqrt{25} = 5$, which is **not** the same as $\sqrt{9} + \sqrt{16} = 3 + 4 = 7$. The root does not distribute over addition.

Mental-Math Shortcut

To find the square root of a **perfect square** between 100 and 9999, recover it digit by digit:

1. **Tens digit** – trap the number between two "round" squares. For $\sqrt{3249}$: since $2500 < 3249 < 3600$, the root is between 50 and 60, so the tens digit is 5.
2. **Units digit** – look at the last digit of the number and ask which digit squared ends that way. 3249 ends in 9, and only $3^2 = 9$ and $7^2 = 49$ end in 9, so the answer is 53 or 57.
3. **Pick** the candidate on the correct side: 3249 is nearer 3600 than 2500, so take the larger, 57. Check: $57^2 = 3249$. ✓

The same routine gives $\sqrt{576}$: between 20 and 30; ends in 6 so 24 or 26; nearer $400 \Rightarrow 24$.

⚠ Common Mistakes

- Writing $\sqrt{9+16} = 3+4 = 7$. You must add first: $\sqrt{25} = 5$. The radical groups everything under it.
- Claiming $\sqrt{16} = \pm 4$. The radical symbol returns only the non-negative (principal) root, 4. The $\pm$ appears when *you* solve $x^2 = 16$, not from the symbol itself.
- Saying $\sqrt{-25} = -5$. No real number squares to a negative; $\sqrt{-25}$ is undefined in the real numbers (whereas $-\sqrt{25} = -5$ is fine).
- Leaving a root unsimplified or simplifying with a non-square factor: $\sqrt{72} = 6\sqrt{2}$, not $\sqrt{8}\sqrt{9}$ left as-is, and not $\sqrt{72} = \sqrt{6}\sqrt{12}$ (neither 6 nor 12 is a perfect square).
- Confusing $\sqrt{a}$ with $a/2$. $\sqrt{36} = 6$, not 18; halving and rooting are different operations.

📝 Quiz

11.1.Q1. Find $\sqrt{81}$.

A. 8

B. 9

C. 18

D. 40.5

11.1.Q2. Between which two consecutive whole numbers does $\sqrt{40}$ lie?

A. 5 and 6

B. 6 and 7

C. 4 and 5

D. 20 and 21

11.1.Q3. Simplify $\sqrt{50}$.

11.1.Q4. Evaluate $\sqrt{9+16}$.

A. 7

B. 5

C. 25

D. 3.5

11.1.Q5. A square field has area $169\ \text{m}^2$. Find its side length.

11.1.Q6. True or False: $\sqrt{-49} = -7$.

A. True

B. False

11.1.Q7. Simplify $\sqrt{12} + \sqrt{27}$.

11.1.Q8. Find $\sqrt[3]{64}$ (cube root).

🐍 From Concept to Code (Python)

Python gives you three handy tools for roots. `x ** 0.5` uses the exponent law $\sqrt{x} = x^{1/2}$; `math.sqrt(x)` is the dedicated function (clearer, and it errors on negatives); and `math.isqrt(n)` returns the exact **integer** square root of a whole number – perfect for checking perfect squares.

```
import math
print(49 ** 0.5)        # square root via exponent
print(math.sqrt(50))    # not a perfect square
print(math.isqrt(225))  # exact integer root
```

```
output
7.0
7.071067811865475
15
```

```
import math
n = 144
# a perfect square: isqrt squared returns n
print(math.isqrt(n) ** 2 == n)
print(math.isqrt(150) ** 2 == 150)
```

```
output
True
False
```

Try it:

1. Try `math.sqrt(-9)`. What error do you get, and why does it match the maths?
2. Write a loop over `range(1, 21)` that prints each `n` and `math.isqrt(n)`, marking the perfect squares where `math.isqrt(n)**2 == n`.

Answer Key – Chapter 11 – Square Roots

11.1.Q1. 9 – because $9^2 = 81$.

11.1.Q2. 6 and 7 – since $36 < 40 < 49$.

11.1.Q3. $5\sqrt{2}$ – because $50 = 25 \times 2$ and $\sqrt{25} = 5$.

11.1.Q4. 5 – add inside first (25), then root.

11.1.Q5. $13\ \text{m}$ – $\sqrt{169} = 13$.

11.1.Q6. False – no real number squares to -49; $\sqrt{-49}$ is undefined in the reals.

11.1.Q7. $5\sqrt{3} - 2\sqrt{3} + 3\sqrt{3}$.

11.1.Q8. 4 – because $4^3 = 64$.

Chapter 12 · Scientific Notation

How to write very large and very small numbers compactly as $a \times 10^n$, convert to and from ordinary form, and multiply or divide them by working on the powers of 10.

The Idea Behind It

Multiplying by 10^n just slides the decimal point n places, that is the whole trick of $a \times 10^n$. *but why?*

Our number system is built on a single rule of place value, every column is worth **ten times** the column on its right:

$$\boxed{\text{each place} = 10 \times (\text{the place to its right})}$$

So multiplying by 10 promotes every digit one column up, units become tens, tenths become units, and on paper the only thing that visibly moves is the point, one step **right**. Watch 6.02 climb, one factor of 10 at a time:

$$6.02 \xrightarrow{\times 10} 60.2 \xrightarrow{\times 10} 602. \xrightarrow{\times 10} 6020.$$

Three tens applied is exactly $\times 10^3$, and the point has shifted 3 places right. Dividing by ten (a **negative** power) runs it the other way, each $\times 10^{-1}$ slides the point one place **left**:

$$6.02 \times 10^3 = 6020\,, \qquad 6.02 \times 10^{-3} = 0.00602$$

$$\therefore\ a \times 10^n = a \text{ with the point moved } n \text{ places } (\text{right if } n > 0,\ \text{left if } n < 0)$$

That is why the exponent alone tells you the size: it is a *place-counter*, not extra arithmetic.

×10 = shove every digit up one column, so the point steps over

The Sun is about $150{,}000{,}000{,}000$ metres away; a hydrogen atom is about 0.0000000001 metres wide. Writing – and reading – that many zeros is error-prone. **Scientific notation** rewrites any number as a single tidy digit-string times a power of 10, so size lives entirely in the exponent. It is the standard language of science, engineering and the calculator display.

SCIENTIFIC NOTATION

A number is in **scientific notation** when it is written as

$$a \times 10^n,$$

where the **coefficient** (or **mantissa**) satisfies $1 \le a < 10$ (exactly one nonzero digit before the decimal point) and the **exponent** n is an integer. A **positive** n marks a large number; a **negative** n marks a small one.

The coefficient carries the significant digits, and the power of 10 does the place-value shifting. Recall from Chapter 10 that multiplying by 10^n moves the decimal point n places **right**, and by 10^{-n} moves it n places **left**.

Converting ordinary form ⇒ scientific notation

1. Place the decimal point just after the **first nonzero digit** to get a.
2. Count how many places the point moved. Moving **left** (big number) gives a **positive** exponent; moving **right** (small number) gives a **negative** exponent.

Converting scientific notation ⇒ ordinary form

Move the decimal point n places: **right** if $n > 0$, **left** if $n < 0$, filling with zeros as needed.

Multiplying and dividing

Handle the coefficients and the powers of 10 separately, using the product and quotient laws of Chapter 10 – **add** exponents to multiply, **subtract** to divide. Then, if the new coefficient has drifted outside $1 \le a < 10$, renormalise by adjusting the power of 10.

$$(a \times 10^m) \times (b \times 10^n) = (a\,b) \times 10^{m+n}, \qquad \frac{a \times 10^m}{b \times 10^n} = \frac{a}{b} \times 10^{m-n}$$

Worked examples

Example 1

Write $93{,}000{,}000$ (miles to the Sun) in scientific notation.

SOLUTION

1. First nonzero digit is 9; put the point after it: $a = 9.3$. *one digit before point*
2. From $93000000.$ the point moved 7 places **left**, and the number is large.
3. So the exponent is $+7$.

9.3×10^7

Example 2

Write 0.00046 in scientific notation.

SOLUTION

1. First nonzero digit is 4; place the point after it: $a = 4.6$.
2. From 0.00046 the point moved 4 places **right**, and the number is small. *small ⇒ negative*
3. So the exponent is -4.

4.6×10^{-4}

Example 3

Convert 6.02×10^5 to ordinary form.

SOLUTION

1. The exponent is $+5$, so move the decimal point 5 places to the **right**. + ⇒ go right
2. Starting from 6.02: $6.02 \to 602000.$, adding zeros to fill the places.

$602{,}000$

Example 4

Convert 2.9×10^{-6} to ordinary form.

SOLUTION

1. The exponent is -6, so move the point 6 places to the **left**. − ⇒ go left
2. Starting from 2.9: place the point six steps left, padding with zeros: 0.0000029.

0.0000029

Example 5

Multiply $(3 \times 10^4) \times (2 \times 10^5)$.

SOLUTION

1. Multiply the coefficients: $3 \times 2 = 6$. numbers ×
2. Add the exponents (product law): $10^{4+5} = 10^9$. powers add
3. Coefficient 6 is already in $[1, 10)$, so no renormalising needed.

6×10^9

Example 6

Multiply $(5 \times 10^3) \times (4 \times 10^6)$ and renormalise.

SOLUTION

1. Coefficients: $5 \times 4 = 20$. Exponents: $10^{3+6} = 10^9$, giving 20×10^9.
2. But 20 is not less than 10. Rewrite $20 = 2.0 \times 10^1$. shift, bump power +1
3. Combine: $2.0 \times 10^1 \times 10^9 = 2.0 \times 10^{10}$.

2×10^{10}

Example 7

Divide $\dfrac{8 \times 10^7}{2 \times 10^3}$.

SOLUTION

1. Divide the coefficients: $8 \div 2 = 4$. *numbers ÷*
2. Subtract the exponents (quotient law): $10^{7-3} = 10^4$. *powers subtract*
3. Coefficient 4 is in range, so leave it.

4×10^4

Example 8

Divide and renormalise: $\dfrac{3 \times 10^4}{6 \times 10^8}$.

SOLUTION

1. Divide the coefficients: $3 \div 6 = 0.5$. *numbers ÷*
2. Subtract the exponents: $10^{4-8} = 10^{-4}$, giving 0.5×10^{-4}.
3. But 0.5 is below 1, so renormalise: write $0.5 = 5 \times 10^{-1}$. *below 1 ⇒ shift right, power −1*
4. Combine the powers: $5 \times 10^{-1} \times 10^{-4} = 5 \times 10^{-5}$.

5×10^{-5}

Example 9

Adding in scientific notation: $(4.2 \times 10^6) + (3 \times 10^5)$.

SOLUTION

1. Unlike $\times$ and $\div$, you cannot just add coefficients – the powers of 10 must **match first**. Rewrite the smaller-exponent term to share the larger exponent 10^6. *line up the powers*
2. $3 \times 10^5 = 0.3 \times 10^6$ (drop the exponent by 1, shift the point left by 1).
3. Now both sit over 10^6, so add the coefficients: $4.2 + 0.3 = 4.5$. *same power ⇒ add tops*
4. Attach the shared power: 4.5×10^6, and the coefficient is already in $[1, 10)$.

4.5×10^6

Example 10

Combined operation. Light travels 3×10^8 m/s. How far does it go in $2{,}000$ seconds? Give the answer in scientific notation.

SOLUTION

1. Distance = speed × time. First write the time in scientific notation: $2000 = 2 \times 10^3$ s. *tidy the time*
2. Multiply the coefficients: $3 \times 2 = 6$. *numbers ×*
3. Add the exponents (product law): $10^{8+3} = 10^{11}$. *powers add*
4. Coefficient 6 is in range, so the distance is 6×10^{11} m.

6×10^{11} m **(about 600 billion metres).**

Note: When you multiply or divide and the coefficient lands outside $1 \le a < 10$, you are not finished – renormalise. If a is 10 or more, shift the point left and add 1 to the exponent (e.g. $20 \times 10^9 = 2 \times 10^{10}$). If a is below 1, shift right and subtract 1 (e.g. $0.4 \times 10^6 = 4 \times 10^5$).

⚡ Mental-Math Shortcut

Read the exponent as a **place-counter** and you can size a number instantly, no zeros drawn.

- A **positive** exponent n means the number is "a followed by enough zeros to fill n places after the leading digit". $7 \times 10^5 \Rightarrow 700000$ (the 7, then 5 places).
- A **negative** exponent $-n$ means "the 1st significant digit sits in the nth place after the decimal point". $7 \times 10^{-5} \Rightarrow 0.00007$ (the 7 is in the 5th place).

To compare two numbers in scientific notation, compare exponents first; only if they tie do you compare coefficients. So $3 \times 10^8 > 9 \times 10^7$ at a glance, because $8 > 7$.

⚠ Common Mistakes

- Leaving a coefficient outside $1 \le a < 10$: 42×10^3 is **not** proper form – it is 4.2×10^4. Exactly one nonzero digit goes before the point.
- Getting the sign of the exponent backwards. Large numbers take a **positive** exponent ($5000 = 5 \times 10^3$); small numbers take a **negative** one ($0.005 = 5 \times 10^{-3}$).
- Miscounting the decimal shift by one: in 0.00046 the point moves 4 places (to land just after the 4), giving 10^{-4}, not 10^{-3}.
- Multiplying the exponents when multiplying the numbers: $(3 \times 10^4)(2 \times 10^5) = 6 \times 10^9$ – you **add** the exponents ($4 + 5$), not multiply them.
- Forgetting to renormalise after arithmetic: $5 \times 4 = 20$, so the answer is 2×10^{10}, not 20×10^9.

Quiz

12.1.Q1. Write $58{,}000$ in scientific notation.

A. 5.8×10^4

B. 58×10^3

C. 5.8×10^3

D. 0.58×10^5

12.1.Q2. Write 0.0072 in scientific notation.

A. 7.2×10^{-3}

B. 7.2×10^3

C. 72×10^{-4}

D. 7.2×10^{-2}

12.1.Q3. Convert 4.1×10^6 to ordinary form.

12.1.Q4. Convert 3×10^{-4} to ordinary form.

12.1.Q5. Compute $(2 \times 10^3) \times (4 \times 10^5)$.

12.1.Q6. Compute $\dfrac{9 \times 10^8}{3 \times 10^2}$.

12.1.Q7. Renormalise 35×10^4.

A. 3.5×10^5

B. 3.5×10^4

C. 35×10^5

D. 0.35×10^6

12.1.Q8. True or False: 6×10^3 is larger than 9×10^2.

A. True

B. False

🐍 From Concept to Code (Python)

Python understands scientific notation directly: a literal like `1e6` means 1×10^6, and `4.6e-4` means 4.6×10^{-4}. To *display* a number in scientific notation, use an f-string format spec: `f"{x:.2e}"` shows it as a coefficient with 2 decimal places times a power of 10.

```
distance = 9.3e7        # 93 million, i.e. 9.3 x 10^7
print(distance)
print(2e3 * 4e5)        # Python adds the exponents for you
print(1.5e-4)           # a small number
```

```
output
93000000.0
800000000.0
0.00015
```

```
atoms = 602000000000000000000000   # Avogadro-sized
print(f"{atoms:.2e}")            # format in scientific notation
print(f"{0.0000046:.1e}")
```

```
output
6.02e+23
4.6e-06
```

Try it:

1. Predict, then run: `print(3e4 * 2e5)`. Does the result match 6×10^9?
2. Use `f"{x:.3e}"` to print the speed of light, `x = 299792458`, in scientific notation.

Answer Key – Chapter 12 – Scientific Notation

12.1.Q1. 5.8×10^4 – point moves 4 places left; coefficient between 1 and 10.

12.1.Q2. 7.2×10^{-3} – small number, point moves 3 places right.

12.1.Q3. $4{,}100{,}000$ – move the point 6 places right.

12.1.Q4. 0.0003 – move the point 4 places left.

12.1.Q5. 8×10^8 – $2 \times 4 = 8$, add exponents $3 + 5$.

12.1.Q6. 3×10^6 – $9 \div 3 = 3$, subtract exponents $8 - 2$.

12.1.Q7. 3.5×10^5 – shift the point one left, add 1 to the exponent.

12.1.Q8. True – $6000 > 900$; compare exponents first ($3 > 2$).

Chapter 13 · Unit Conversions

How to convert between units reliably using the unit-fraction (factor-label) method – multiplying by carefully chosen fractions equal to 1 so the unwanted units cancel.

The Idea Behind It

Converting units never changes the amount, because every conversion fraction secretly equals 1.

but why?

Start from a fact you already accept, that 100 centimetres *is* one metre, the very same length wearing a different label:

$$100\ \text{cm} = 1\ \text{m}$$

Divide both sides by $1\ \text{m}$. Equal quantities over the same thing give a fraction worth exactly **one**, the rule the whole method leans on:

$$\boxed{\frac{100\ \text{cm}}{1\ \text{m}} = 1}$$

And multiplying anything by 1 leaves it untouched. So to recast $2\ \text{m}$ in centimetres, multiply by this special 1 and let the matching unit labels cancel like common factors:

$$2\ \text{m} \times \frac{100\ \text{cm}}{1\ \text{m}} = 2 \times 100\ \text{cm} = 200\ \text{cm}$$

The "m" on top and bottom cancel, the number scales, and because we only ever multiplied by 1 the *length itself* is identical:

$$\therefore\ 2\ \text{m} = 200\ \text{cm} \quad (\text{same length, relabelled})$$

That is the safety net of the factor-label method: pick the fraction so the unwanted unit sits underneath and cancels, and the arithmetic, multiply or divide, is decided **for** you.

multiply by 1, the amount cannot change, only its units do

"Add 250 grams of flour", "drive 90 kilometres", "the cell is 2 microns wide" – measurement only means something once we know the **unit**. Converting between units ($\text{cm} \to \text{m}$, $\text{hours} \to \text{seconds}$, $\text{microns} \to \text{picometres}$) is one of the most useful everyday skills, and there is a single method that never lets you guess whether to multiply or divide: the **factor-label method**.

UNIT FRACTION (CONVERSION FACTOR)

A **unit fraction** is a fraction whose top and bottom are equal quantities written in different units, so its value is exactly 1. For example, since $1\ \text{m} = 100\ \text{cm}$:

$$\frac{100\ \text{cm}}{1\ \text{m}} = 1 \qquad \text{and} \qquad \frac{1\ \text{m}}{100\ \text{cm}} = 1.$$

Multiplying by 1 never changes the *amount* – only the units it is expressed in.

The factor-label method works because units cancel like numbers in a fraction. You choose each unit fraction so the unit you want to remove sits **opposite** to where it currently is, and cancels:

$$\text{quantity} \times \frac{\text{wanted unit}}{\text{given unit}} = \text{answer in wanted unit}$$

The metric prefixes

The metric (SI) system scales by powers of 10, which makes its conversion factors clean. The common length prefixes, from large to small:

Prefix	Symbol	Meaning	In metres
kilo	k	10^{3}	$1\ \text{km} = 1000\ \text{m}$
(base)	m	10^{0}	$1\ \text{m}$
centi	c	10^{-2}	$1\ \text{cm} = 0.01\ \text{m}$
milli	m	10^{-3}	$1\ \text{mm} = 0.001\ \text{m}$
micro	μ	10^{-6}	$1\ \mu\text{m} = 10^{-6}\ \text{m}$
nano	n	10^{-9}	$1\ \text{nm} = 10^{-9}\ \text{m}$
pico	p	10^{-12}	$1\ \text{pm} = 10^{-12}\ \text{m}$

The recipe

1. Write the starting quantity as a fraction over 1.
2. Multiply by a unit fraction whose **bottom** matches the unit you want to cancel.
3. Repeat (chain factors) until only the target unit remains.
4. Multiply the numbers across the top, divide by the numbers across the bottom.

Worked examples

Example 1

Convert $5\ \text{km}$ to metres.

SOLUTION

1. Use the unit fraction $\frac{1000\ \text{m}}{1\ \text{km}}$ – km on the bottom so it cancels. *km on bottom*
2. $5\ \text{km} \times \frac{1000\ \text{m}}{1\ \text{km}}$ – the "km" cancel, leaving metres.
3. Multiply: $5 \times 1000 = 5000$.

$5\ \text{km} = 5000\ \text{m}$

Example 2

Convert $250\ \text{cm}$ to metres.

SOLUTION

1. Want to cancel cm, so use $\dfrac{1\ \text{m}}{100\ \text{cm}}$ (cm on the bottom). *cm on bottom*
2. $250\ \text{cm} \times \dfrac{1\ \text{m}}{100\ \text{cm}}$ – cm cancels.
3. Divide: $250 \div 100 = \boxed{2.5}$.

$250\ \text{cm} = \boxed{2.5\ \text{m}}$

Example 3

Convert $2.5\ \text{hours}$ to seconds by chaining two factors.

SOLUTION

1. Hours $\to$ minutes: $\times \dfrac{60\ \text{min}}{1\ \text{h}}$.
2. Minutes $\to$ seconds: $\times \dfrac{60\ \text{s}}{1\ \text{min}}$.
3. Chain them: $2.5\ \text{h} \times \dfrac{60\ \text{min}}{1\ \text{h}} \times \dfrac{60\ \text{s}}{1\ \text{min}}$ – both "h" and "min" cancel. *chain the factors*
4. Multiply across: $2.5 \times 60 \times 60 = \boxed{9000}$.

$2.5\ \text{h} = \boxed{9000\ \text{s}}$

Example 4

The E. coli example. An *E. coli* bacterium is about $2\ \mu\text{m}$ (microns) long. Express that length in picometres.

SOLUTION

1. Convert in two clean steps via metres. First $\mu\text{m} \to \text{m}$: $1\ \mu\text{m} = 10^{-6}\ \text{m}$.
2. Then $\text{m} \to \text{pm}$: $1\ \text{pm} = 10^{-12}\ \text{m}$, so the unit fraction is $\dfrac{1\ \text{pm}}{10^{-12}\ \text{m}}$.
3. Chain: $2\ \mu\text{m} \times \dfrac{10^{-6}\ \text{m}}{1\ \mu\text{m}} \times \dfrac{1\ \text{pm}}{10^{-12}\ \text{m}}$ – "µm" and "m" cancel. *via metres*
4. Combine the powers of 10 (Chapter 10): $\dfrac{10^{-6}}{10^{-12}} = 10^{\boxed{-6-(-12)}} = \boxed{10^{6}}$, so 2×10^{6}.

subtract the powers

$2\ \mu\text{m} = 2{,}000{,}000\ \text{pm} = \boxed{2 \times 10^{6}\ \text{pm}}$

Example 5

Convert a speed of $90\ \mathrm{km/h}$ to metres per second.

SOLUTION

1. Two conversions act at once: km $\to$ m on top, h $\to$ s on the bottom. *fix both parts*
2. $\dfrac{90\ \mathrm{km}}{1\ \mathrm{h}} \times \dfrac{1000\ \mathrm{m}}{1\ \mathrm{km}} \times \dfrac{1\ \mathrm{h}}{3600\ \mathrm{s}}$ – "km" and "h" cancel.
3. Numbers: $\dfrac{90 \times 1000}{3600} = \dfrac{90000}{3600} = 25$.

$90\ \mathrm{km/h} = 25\ \mathrm{m/s}$

Example 6

A recipe needs $1.5\ \mathrm{kg}$ of sugar; the bag is sold in grams. How many grams?

SOLUTION

1. Cancel kg with $\dfrac{1000\ \mathrm{g}}{1\ \mathrm{kg}}$. *kg on bottom*
2. $1.5\ \mathrm{kg} \times \dfrac{1000\ \mathrm{g}}{1\ \mathrm{kg}}$ – kg cancels.
3. Multiply: $1.5 \times 1000 = 1500$.

$1.5\ \mathrm{kg} = 1500\ \mathrm{g}$

Example 7

A squared unit. A floor is $3\ \mathrm{m}^2$. Convert that area to square centimetres.

SOLUTION

1. The length factor is $1\ \mathrm{m} = 100\ \mathrm{cm}$, but area uses **two** lengths, so the factor must be **squared**. Use $\left(\dfrac{100\ \mathrm{cm}}{1\ \mathrm{m}}\right)^2$. *square the factor*
2. Squaring top and bottom: $\left(\dfrac{100\ \mathrm{cm}}{1\ \mathrm{m}}\right)^2 = \dfrac{10000\ \mathrm{cm}^2}{1\ \mathrm{m}^2}$, and the m^2 cancels.
3. Multiply: $3 \times 10000 = 30000$. *×100², not ×100*

$3\ \mathrm{m}^2 = 30{,}000\ \mathrm{cm}^2$ – **a factor of** $100^2 = 10000$**, not** 100**.**

Example 8

Three factors chained. How many seconds are there in 3 days?

SOLUTION

1. Step from days to seconds one unit at a time, each as a unit fraction: days $\to$ hours $\to$ minutes $\to$ seconds.
2. Chain them so every unwanted unit cancels: $3\ \text{days} \times \dfrac{24\ \text{h}}{1\ \text{day}} \times \dfrac{60\ \text{min}}{1\ \text{h}} \times \dfrac{60\ \text{s}}{1\ \text{min}}$ – "day", "h" and "min" all cancel. *cancel the chain*
3. Multiply across the top: $3 \times 24 \times 60 \times 60 = 259200$.

$3\ \text{days} = 259{,}200\ \text{s}$

Example 9

Converting a price rate. Fuel costs 1.60 dollars per litre. If $1\ \text{gallon} = 3.8\ \text{litres}$, what is the price per gallon?

SOLUTION

1. The rate is $\dfrac{1.60\ \text{dollars}}{1\ \text{litre}}$; we want dollars per **gallon**, so multiply by a factor with litres on top to cancel the litre underneath. Use $\dfrac{3.8\ \text{litres}}{1\ \text{gallon}}$. *litres on top to cancel*
2. $\dfrac{1.60\ \text{dollars}}{1\ \text{litre}} \times \dfrac{3.8\ \text{litres}}{1\ \text{gallon}}$ – "litre" cancels, leaving dollars per gallon.
3. Multiply: $1.60 \times 3.8 = 6.08$.

About 6.08 dollars per gallon.

Note: Always write the units in every line and cancel them like algebra. If your leftover unit is wrong (say you end up with km^2/m instead of m), you used a factor upside down – flip it. The units are a built-in error-checker: get them right and the numbers almost always follow.

Mental-Math Shortcut

Within the **metric system**, converting is just sliding the decimal point by the difference in the powers of 10 – no calculator needed.

- Going to a **smaller** unit (m → cm, km → m), the *number gets bigger*: move the point **right**. $3\ \text{km} \to 3000\ \text{m}$ (3 places, since km is 10^3 m).
- Going to a **larger** unit (g → kg, mm → m), the *number gets smaller*: move the point **left**. $450\ \text{g} \to 0.45\ \text{kg}$.

For prefixes far apart, just **subtract the exponents**: micro (10^{-6}) to pico (10^{-12}) is a jump of $-6 - (-12) = 6$ places, so multiply by 10^6 – exactly the E. coli example.

⚠ Common Mistakes

- Multiplying when you should divide (or vice versa). The factor-label method removes the guesswork: pick the unit fraction so the old unit **cancels**, then the arithmetic is forced.
- Inverting the conversion factor: to go $\text{cm} \to \text{m}$ you multiply by $\frac{1\ \text{m}}{100\ \text{cm}}$ (÷100), not $\frac{100\ \text{cm}}{1\ \text{m}}$ (×100). Check that the unwanted unit is on the bottom.
- Mixing prefixes carelessly: $1\ \mu\text{m} = 10^{-6}\ \text{m}$ and $1\ \text{nm} = 10^{-9}\ \text{m}$, so $1\ \mu\text{m} = 1000\ \text{nm}$ – a factor of 1000, not 10.
- Forgetting to convert **both** parts of a compound unit: km/h → m/s needs the km→m factor *and* the h→s factor; doing only one gives a nonsense answer.
- For areas/volumes, forgetting the factor is squared/cubed: $1\ \text{m}^2 = 100^2 = 10000\ \text{cm}^2$, not $100\ \text{cm}^2$.

Quiz

13.1.Q1. Convert $3\ \text{km}$ to metres.

A. $30\ \text{m}$

B. $300\ \text{m}$

C. $3000\ \text{m}$

D. $0.003\ \text{m}$

13.1.Q2. Convert $750\ \text{cm}$ to metres.

A. $7.5\ \text{m}$

B. $75\ \text{m}$

C. $0.75\ \text{m}$

D. $7500\ \text{m}$

13.1.Q3. Convert $4\ \text{hours}$ to seconds.

13.1.Q4. An object is $5\ \mu\text{m}$ long. How many picometres is that?

13.1.Q5. Convert $2\ \text{kg}$ to grams.

13.1.Q6. Convert $72\ \text{km/h}$ to metres per second.

13.1.Q7. Which unit fraction converts millimetres to metres?

A. $\dfrac{1000\ \text{mm}}{1\ \text{m}}$

B. $\dfrac{1\ \text{m}}{1000\ \text{mm}}$

C. $\dfrac{1\ \text{m}}{100\ \text{mm}}$

D. $\dfrac{1\ \text{mm}}{1000\ \text{m}}$

13.1.Q8. True or False: $1\ \mu\text{m} = 1000\ \text{nm}$.

A. True

B. False

🐍 From Concept to Code (Python)

Conversions are a perfect first use of **functions** and **named constants**. Store each conversion factor as a clearly named variable, then write a tiny function that multiplies by it – the code reads like the maths, and you reuse it freely.

```
# named constants make the factors self-documenting
M_PER_KM = 1000

def km_to_m(km):
    return km * M_PER_KM

print(km_to_m(5))      # 5 km
print(km_to_m(2.5))    # works for decimals too
```

```
output
5000
2500.0
```

```
# the E. coli micron -> picometre conversion
def micron_to_pm(um):
    return um * 1e6        # micro (1e-6 m) to pico (1e-12 m): factor 1e6

print(micron_to_pm(2))
print(f"{micron_to_pm(2):.1e}")
```

```
output
2000000.0
2.0e+06
```

Try it:

1. Write a function `cm_to_m(cm)` that returns `cm / 100`, and test it on `250`.
2. Write `kmh_to_ms(speed)` that converts km/h to m/s (multiply by `1000/3600`), then check that `kmh_to_ms(90)` gives `25.0`.

Answer Key – Chapter 13 – Unit Conversions

13.1.Q1. $3000\ \text{m}$ – $\times \dfrac{1000\ \text{m}}{1\ \text{km}}$.

13.1.Q2. $7.5\ \text{m}$ – divide by 100.

13.1.Q3. $14{,}400\ \text{s}$ – $4 \times 60 \times 60$.

13.1.Q4. $5 \times 10^6\ \text{pm}$ ($5{,}000{,}000\ \text{pm}$) – micro to pico is a jump of 10^6.

13.1.Q5. $2000\ \text{g}$ – $\times \dfrac{1000\ \text{g}}{1\ \text{kg}}$.

13.1.Q6. $20\ \text{m/s}$ – $\dfrac{72 \times 1000}{3600}$.

13.1.Q7. $\dfrac{1\ \text{m}}{1000\ \text{mm}}$ – mm on the bottom so it cancels.

13.1.Q8. True – $10^{-6}\ \text{m}$ vs $10^{-9}\ \text{m}$ differ by a factor of 1000.



PART IV

Fractions, Decimals & Percentages

Three faces of the same idea – parts of a whole. Operating on fractions and decimals, converting fluently between forms, and working with percentages.

Chapter 14 – Fractions · Chapter 15 – Decimals · Chapter 16 – Percentages

Chapter 14 · Fractions

How to read, simplify and compare fractions, and add, subtract, multiply and divide them – including converting between mixed numbers and improper fractions.

The Idea Behind It

Dividing by a fraction is the same as multiplying by its reciprocal: $\frac{1}{2} \div \frac{3}{4} = \frac{1}{2} \times \frac{4}{3}$. *but why?*

"Flip and multiply" feels like a trick pulled from nowhere. It is really just the rule that **multiplying top and bottom by the same nonzero number leaves a fraction unchanged**, applied once cleverly. The key fact is that a fraction times its **reciprocal** is 1:

$$\boxed{\frac{c}{d} \times \frac{d}{c} = \frac{cd}{dc} = 1}$$

Write the division $a \div \frac{c}{d}$ as one tall fraction, then multiply its top and bottom by $\frac{d}{c}$ (legal, since that is multiplying by 1). The bottom collapses to 1 and the division disappears:

$$a \div \frac{c}{d} = \frac{a}{\frac{c}{d}} = \frac{a \times \frac{d}{c}}{\frac{c}{d} \times \frac{d}{c}} = \frac{a \times \frac{d}{c}}{1} = a \times \frac{d}{c}$$

The denominator became 1 for exactly the reason above, so all that survives is a times the **flipped** fraction. Test it on $\frac{1}{2} \div \frac{3}{4}$:

$$\frac{1}{2} \div \frac{3}{4} = \frac{1}{2} \times \frac{4}{3} = \frac{4}{6} = \frac{2}{3}$$

$$\therefore \; a \div \frac{c}{d} = a \times \frac{d}{c}$$

So "invert and multiply" is not a shortcut to memorise blindly: it is forced by the fact that a fraction and its reciprocal cancel to 1. *flip the second, keep the first*

A **fraction** is a number that names part of a whole, or the result of a division that has not been carried out. When you split one pizza equally among four people, each share is $\frac{1}{4}$ of the pizza. The same symbol $\frac{1}{4}$ also means "1 divided by 4". Both readings are correct, and keeping them in mind makes every rule below feel natural rather than arbitrary.

FRACTION

A fraction $\frac{a}{b}$ (with $b \neq 0$) has two parts: - the **numerator** a on top – *how many* parts you have; - the **denominator** b on the bottom – how many *equal parts* the whole is cut into.

The bar itself means "divided by", so $\frac{a}{b} = a \div b$.

A fraction is **proper** when the top is smaller than the bottom (its value is less than 1, like $\frac{3}{8}$), and **improper** when the top is at least as big as the bottom (its value is 1 or more, like $\frac{7}{4}$). A **mixed number** such as $1\frac{3}{4}$ is a whole number written alongside a proper fraction.

Equivalent fractions and simplifying

Multiplying *or* dividing the top and bottom by the **same** non-zero number does not change a fraction's value – you are only re-cutting the same amount into more, or fewer, pieces. Fractions that look different but name the same number are **equivalent**.

$$\frac{a}{b} = \frac{a \times k}{b \times k} = \frac{a \div k}{b \div k} \qquad (k \neq 0)$$

To **simplify** (or *reduce*) a fraction, divide top and bottom by their greatest common factor until no factor other than 1 remains. A fraction in this state is in its **lowest terms**.

The four operations

- **Multiply:** multiply the tops and multiply the bottoms – $\frac{a}{b} \times \frac{c}{d} = \frac{ac}{bd}$.
- **Divide:** *invert and multiply* – turn the second fraction upside-down (its **reciprocal**) and multiply: $\frac{a}{b} \div \frac{c}{d} = \frac{a}{b} \times \frac{d}{c}$.
- **Add / subtract:** you can only combine parts of the *same size*, so first rewrite both fractions over a **common denominator**, then add or subtract the numerators and keep the denominator.

$$\frac{a}{b} + \frac{c}{d} = \frac{ad}{bd} + \frac{cb}{bd} = \frac{ad + cb}{bd}$$

Worked examples

Example 1

Simplify $\frac{18}{24}$ to lowest terms.

SOLUTION

1. Find the greatest common factor of 18 and 24. Both divide by 6 (since $18 = 6 \times 3$ and $24 = 6 \times 4$).

   biggest shared factor

2. Divide top and bottom by the shaded 6: $\frac{18 \div 6}{24 \div 6} = \frac{3}{4}$. ÷ top and bottom
3. Check: 3 and 4 share no common factor beyond 1, so it is fully reduced.

$\frac{18}{24} = \frac{3}{4}$

Example 2

Multiply $\frac{3}{4} \times \frac{8}{9}$.

SOLUTION

1. Multiply tops and bottoms straight across: $\frac{3 \times 8}{4 \times 9} = \frac{24}{36}$. *no common denom needed*
2. Simplify by dividing top and bottom by 12: $\frac{24 \div 12}{36 \div 12} = \frac{2}{3}$.
3. Shortcut: you could *cancel before multiplying* – $\frac{3}{9} = \frac{1}{3}$ and $\frac{8}{4} = \frac{2}{1}$ give $\frac{1 \times 2}{3 \times 1} = \frac{2}{3}$ directly.

cancel first = less reducing

$\frac{2}{3}$

Example 3

Divide $\frac{5}{6} \div \frac{2}{3}$.

SOLUTION

1. Invert the **second** fraction: the reciprocal of $\frac{2}{3}$ is $\frac{3}{2}$, and the $\div$ becomes a $\times$.

flip the 2nd, keep the 1st

2. Multiply: $\frac{5}{6} \times \frac{3}{2} = \frac{15}{12}$.
3. Simplify by dividing top and bottom by 3: $\frac{15 \div 3}{12 \div 3} = \frac{5}{4}$.

$\frac{5}{4}$ **(the same as $1\frac{1}{4}$)**

Example 4

Add $\frac{2}{3} + \frac{1}{4}$.

SOLUTION

1. The denominators are 3 and 4; their least common multiple is 12. *common denominator first*
2. Rewrite each fraction over 12: $\frac{2}{3} = \frac{8}{12}$ and $\frac{1}{4} = \frac{3}{12}$.
3. Now the parts are the same size, so add the numerators only: $\frac{8 + 3}{12} = \frac{11}{12}$. *keep the bottom*

$\frac{11}{12}$

Example 5

Subtract $\frac{7}{10} - \frac{2}{15}$.

SOLUTION

1. Find a common denominator for 10 and 15: the least common multiple is 30.
2. Rewrite each over 30: $\frac{7}{10} = \frac{21}{30}$ and $\frac{2}{15} = \frac{4}{30}$. *scale both up*
3. Subtract numerators only: $\frac{21 - 4}{30} = \frac{17}{30}$, which is already in lowest terms.

$\frac{17}{30}$

Example 6

Convert $\frac{23}{5}$ to a mixed number, then back again.

SOLUTION

1. Divide 23 by 5: it goes in 4 times with 3 left over, since $5 \times 4 = 20$ and $23 - 20 = 3$.
   how many whole 5s?
2. The whole part 4 and remainder 3 sit over the original denominator: $\frac{23}{5} = 4\frac{3}{5}$.
3. To reverse it, multiply the whole by the denominator and add the numerator: $4 \times 5 + 3 = 23$, giving back $\frac{23}{5}$. *whole·bottom + top*

$\frac{23}{5} = 4\frac{3}{5}$

Example 7

Add the mixed numbers $2\frac{1}{2} + 1\frac{3}{4}$.

SOLUTION

1. Convert each to an improper fraction: $2\frac{1}{2} = \frac{5}{2}$ and $1\frac{3}{4} = \frac{7}{4}$. *mixed → improper*
2. Use the common denominator 4: $\frac{5}{2} = \frac{10}{4}$, so the sum is $\frac{10}{4} + \frac{7}{4} = \frac{17}{4}$.
3. Convert back to a mixed number: $17 \div 4 = 4$ remainder 1, giving $4\frac{1}{4}$. *improper → mixed*

$4\frac{1}{4}$

Example 8

Subtract the mixed numbers $5\frac{1}{6} - 2\frac{3}{4}$.

SOLUTION

1. Turn both into improper fractions: $5\frac{1}{6} = \frac{31}{6}$ (since $5 \times 6 + 1 = 31$) and $2\frac{3}{4} = \frac{11}{4}$. mixed → improper
2. Common denominator of 6 and 4 is 12. Rewrite: $\frac{31}{6} = \frac{62}{12}$ and $\frac{11}{4} = \frac{33}{12}$. scale both to /12
3. Subtract numerators only: $\frac{62 - 33}{12} = \frac{29}{12}$. keep the bottom
4. Convert back: $29 \div 12 = 2$ remainder 5, so $2\frac{5}{12}$. improper → mixed

$5\frac{1}{6} - 2\frac{3}{4} = 2\frac{5}{12}$

Example 9

A combined expression: evaluate $\frac{2}{3} + \frac{1}{2} \times \frac{3}{5}$.

SOLUTION

1. Order of operations (Chapter 1) applies to fractions too: do the **multiplication before the addition**.
   × before +
2. Multiply first – no common denominator needed, just tops and bottoms: $\frac{1}{2} \times \frac{3}{5} = \frac{3}{10}$.
3. Now add $\frac{2}{3} + \frac{3}{10}$ over the common denominator $\text{lcm}(3, 10) = 30$: $\frac{2}{3} = \frac{20}{30}$ and $\frac{3}{10} = \frac{9}{30}$.
4. Add numerators: $\frac{20 + 9}{30} = \frac{29}{30}$, already in lowest terms.

$\frac{29}{30}$

Example 10

A word problem: a tank is $\frac{3}{4}$ full, and you drain $\frac{2}{3}$ of **what is in it**. What fraction of the whole tank did you drain, and how much is left?

SOLUTION

1. "$\frac{2}{3}$ of $\frac{3}{4}$" means **multiply**: $\frac{2}{3} \times \frac{3}{4}$. "of" ⇒ ×
2. Multiply across and reduce: $\frac{2 \times 3}{3 \times 4} = \frac{6}{12} = \frac{1}{2}$. So you drained $\frac{1}{2}$ of the whole tank.
3. The amount left is the starting $\frac{3}{4}$ minus the drained $\frac{1}{2}$, over the common denominator 4: $\frac{3}{4} - \frac{2}{4} = \frac{1}{4}$.
   subtract over /4

You drained $\frac{1}{2}$ of the tank, leaving $\frac{1}{4}$ full.

Note: To multiply, you do **not** need a common denominator – that requirement applies only to addition and subtraction. Mixing the two rules up is the single most common slip with fractions, so name the operation first, then apply its rule.

Tip: Whenever an answer is an improper fraction or is not in lowest terms, finish the job: reduce it and, if the question used mixed numbers, convert back to a mixed number.

numerator denominator equivalent fractions reciprocal improper fraction

⚡ Mental-Math Shortcut

Memorise the common **fraction ↔ percent** partners and conversions become instant:

- $\frac{1}{2} = 50\%$, $\frac{1}{4} = 25\%$, $\frac{3}{4} = 75\%$, $\frac{1}{5} = 20\%$, $\frac{1}{10} = 10\%$, $\frac{1}{3} \approx 33\%$, $\frac{1}{8} = 12.5\%$.

To turn **any** fraction into a percent, divide the top by the bottom and multiply by 100; to go back, write the percent over 100 and simplify. So $\frac{3}{8} = 3 \div 8 = 0.375 = 37.5\%$.

To **compare** two fractions without a full common denominator, **cross-multiply**: for $\frac{a}{b}$ versus $\frac{c}{d}$, compare $a \times d$ against $c \times b$. The bigger cross-product sits above the bigger fraction. For $\frac{3}{5}$ vs $\frac{5}{8}$: $3 \times 8 = 24$ and $5 \times 5 = 25$, so $\frac{5}{8}$ is the larger.

⚠ Common Mistakes

- Adding straight across – writing $\frac{1}{2} + \frac{1}{3} = \frac{2}{5}$. You may only add numerators **over a common denominator**: the real answer is $\frac{3}{6} + \frac{2}{6} = \frac{5}{6}$.
- Finding a common denominator before **multiplying**. Multiplication needs no common denominator – just multiply tops and bottoms: $\frac{2}{3} \times \frac{1}{4} = \frac{2}{12} = \frac{1}{6}$.
- Inverting the **wrong** fraction (or both) when dividing. Keep the first fraction as it is and flip only the **second**: $\frac{a}{b} \div \frac{c}{d} = \frac{a}{b} \times \frac{d}{c}$.
- Cancelling across a $+$ or $-$ sign, e.g. "cancelling" the 3s in $\frac{3+4}{3}$. You may only cancel a factor common to the **whole** top and **whole** bottom, never one term of a sum.
- Leaving the answer unsimplified or as an improper fraction when a mixed number was expected – $\frac{8}{6}$ should be written $\frac{4}{3} = 1\frac{1}{3}$.

Quiz

14.1.Q1. Simplify $\frac{12}{30}$ to lowest terms.

A. $\frac{2}{5}$

B. $\frac{6}{15}$

C. $\frac{1}{3}$

D. $\frac{4}{10}$

14.1.Q2. Compute $\frac{3}{8} \times \frac{4}{9}$.

A. $\frac{12}{72}$

B. $\frac{1}{6}$

C. $\frac{7}{17}$

D. $\frac{12}{17}$

14.1.Q3. Compute $\frac{5}{6} \div \frac{10}{3}$.

14.1.Q4. Compute $\frac{3}{4} + \frac{5}{6}$.

14.1.Q5. Compute $\frac{7}{8} - \frac{1}{2}$.

A. $\frac{3}{8}$

B. $\frac{6}{6}$

C. $\frac{1}{4}$

D. $\frac{6}{8}$

14.1.Q6. Write $\frac{17}{4}$ as a mixed number.

14.1.Q7. Write $3\frac{2}{5}$ as an improper fraction.

14.1.Q8. Which is larger, $\frac{4}{7}$ or $\frac{5}{9}$?

A. $\frac{4}{7}$

B. $\frac{5}{9}$

C. they are equal

D. cannot tell

14.1.Q9. True or False: $\frac{2}{3}$ and $\frac{6}{9}$ are equivalent.

A. True

B. False

🐍 From Concept to Code (Python)

Python's built-in **`fractions.Fraction`** type stores exact fractions – no rounding, and it simplifies automatically. It is the right tool whenever you need precise fraction arithmetic instead of approximate decimals.

```
from fractions import Fraction

a = Fraction(1, 2) + Fraction(1, 3)   # exact, auto-simplified
b = Fraction(3, 4) / Fraction(2, 3)   # divide = invert & multiply
print(a)
print(b)
print(Fraction(18, 24))               # reduces on creation
```

```
output
5/6
9/8
3/4
```

```
from fractions import Fraction

f = Fraction(23, 5)
print(f.numerator, f.denominator)     # the two parts
print(23 // 5, 23 % 5)                 # whole part, remainder -> 4 3/5
print(float(f))                        # as an approximate decimal
```

```
output
23 5
4 3
4.6
```

Try it:

1. Use `Fraction` to confirm that `Fraction(2,3) + Fraction(1,4)` equals `Fraction(11,12)`.
2. Compute `Fraction(5,6) - Fraction(2,15)` and print it; check it matches the chapter answer $\frac{17}{30}$.

Answer Key – Chapter 14 – Fractions

14.1.Q1. $\frac{2}{5}$ – divide top and bottom by their GCF, 6.

14.1.Q2. $\frac{1}{6}$ – $\frac{12}{72}$ reduces by 12 to $\frac{1}{6}$.

14.1.Q3. $\frac{1}{4}$ – invert and multiply: $\frac{5}{6} \times \frac{3}{10} = \frac{15}{60} = \frac{1}{4}$.

14.1.Q4. $\frac{19}{12}$ (i.e. $1\frac{7}{12}$) – over 12: $\frac{9}{12} + \frac{10}{12}$.

14.1.Q5. $\frac{3}{8}$ – rewrite $\frac{1}{2} = \frac{4}{8}$, then $\frac{7}{8} - \frac{4}{8} = \frac{3}{8}$.

14.1.Q6. $4\frac{1}{4}$ – $17 \div 4 = 4$ remainder 1.

14.1.Q7. $\frac{17}{5}$ – $3 \times 5 + 2 = 17$.

14.1.Q8. $\frac{4}{7}$ – cross-multiply: $4 \times 9 = 36 > 35 = 5 \times 7$.

14.1.Q9. True – multiplying top and bottom of $\frac{2}{3}$ by 3 gives $\frac{6}{9}$.

Chapter 15 · Decimals

How to read decimal place value, add, subtract, multiply and divide decimals, round them sensibly, and convert between decimals and fractions.

The Idea Behind It

Why you must line up the decimal points before adding: $0.3 + 0.07$ is *not* 0.10. *but why?*

Lining up the points is not a tidiness rule, it is the same "you can only add parts of the same size" idea from fractions. Each place to the right of the point is the next **negative power of ten**, ten times smaller than the place before it:

$$\boxed{0.1 = \frac{1}{10} = 10^{-1}, \quad 0.01 = \frac{1}{100} = 10^{-2}, \quad 0.001 = 10^{-3}}$$

So a tenth and a hundredth are **different-sized pieces**. Read 0.3 and 0.07 as the fractions they stand for:

$$0.3 = \frac{3}{10}, \qquad 0.07 = \frac{7}{100}$$

Adding fractions needs a **common denominator** (Chapter 14): rewrite the tenths as hundredths, $\frac{3}{10} = \frac{30}{100}$, then the parts match and you add the numerators:

$$\frac{30}{100} + \frac{7}{100} = \frac{37}{100} = 0.37$$

$$\therefore \; 0.3 + 0.07 = 0.37, \text{ not } 0.10$$

Stacking the numbers with their points aligned puts tenths under tenths and hundredths under hundredths automatically, so column addition is doing the common-denominator step for you.

align the points = match the place values

A **decimal** extends our ordinary place-value system to the right of the decimal point so we can write amounts smaller than one without fractions. Money is the everyday example: 3.25 dollars means three whole units plus 25 hundredths. Each step to the right makes a place **ten times smaller**, exactly mirroring how each step to the left makes it ten times larger.

DECIMAL PLACE VALUE

Reading rightward from the point, the columns are: - **tenths** $= \frac{1}{10}$ - **hundredths** $= \frac{1}{100}$ - **thousandths** $= \frac{1}{1000}$, and so on.

So $0.426 = \frac{4}{10} + \frac{2}{100} + \frac{6}{1000}$.

A key fact: putting **zeros on the right-hand end** of a decimal never changes its value, because they add nothing. Thus $0.5 = 0.50 = 0.500$. This freedom is what lets us line numbers up for addition and subtraction.

Adding and subtracting

Write the numbers in a column with their **decimal points aligned** – which automatically lines up tenths under tenths, hundredths under hundredths. Pad with trailing zeros so every number has the same number of decimal places, then add or subtract exactly as with whole numbers, bringing the point straight down.

Multiplying

Ignore the points and multiply the numbers as whole numbers. Then count the **total** number of decimal places in the two factors, and place the point in the answer so it has that many decimal places.

$$0.6 \times 0.04 = 24 \text{ (digits)} \rightarrow 0.024 \quad (1 + 2 = 3 \text{ decimal places})$$

Dividing

Never divide by a decimal. **Shift the point** in the divisor to the right until it is a whole number, and shift the point in the dividend by the *same* number of places (multiplying both by the same power of ten leaves the quotient unchanged). Then do ordinary long division, keeping the point in the answer directly above its place in the dividend.

Rounding

To round to a given place, look at the **next digit** to the right. If it is 5 or more, round the target digit **up**; if it is 4 or less, leave it as it is. Drop everything after the target place.

Worked examples

Example 1

Compute $12.4 + 3.76$.

SOLUTION

1. Align the points and pad: write 12.40 above 3.76 so both have two decimal places. *pad with a 0*
2. Add column by column from the right: $0 + 6 = 6$ hundredths; $4 + 7 = 11$ tenths (write 1, carry 1); $2 + 3 + 1 = 6$ ones; $1 + 0 = 1$ ten. ← *carry the 1*
3. Bring the point straight down between the ones and tenths: 16.16.

$12.4 + 3.76 = 16.16$

Example 2

Compute $8 - 2.35$.

SOLUTION

1. Write 8 as $8.\,0\;\;0$ so both numbers have two decimal places, above 2.35. *8 = 8.00*
2. Subtract with borrowing: $0 - 5$ needs a borrow, giving $10 - 5 = 5$ hundredths; the tenths become $9 - 3 = 6$; the ones $7 - 2 = 5$. *borrow across the zeros*
3. Place the point straight down: 5.65.

$8 - 2.35 = 5.65$

Example 3

Compute 1.2×0.05.

SOLUTION

1. Ignore the points and multiply the whole numbers: $12 \times 5 = 60$. *drop the points first*
2. Count decimal places in the factors: 1.2 has 1 and 0.05 has 2, for a total of 3. *1 + 2 = 3 places*
3. Place the point so the answer has 3 decimal places: $60 \to 0.060$.

$1.2 \times 0.05 = 0.06$

Example 4

Compute $4.5 \div 0.9$.

SOLUTION

1. Make the divisor whole: shift the point in 0.9 one place right to get 9. *×10 to clear the point*
2. Shift the dividend the **same** one place: $4.5 \to 45$. The problem is now $45 \div 9$. *shift both equally*
3. Divide: $45 \div 9 = 5$.

$4.5 \div 0.9 = 5$

Example 5

Round 3.14159 to two decimal places.

SOLUTION

1. The second decimal place is the 4 (the hundredths). Look at the next digit to its right: 1. *this digit decides*
2. Since $1 \leq 4$, the 4 stays unchanged. *4 or less → round down*
3. Drop everything after the hundredths place, leaving 3.14.

$3.14159 \approx 3.14$

Example 6

Convert $\frac{3}{8}$ to a decimal, and 0.35 to a fraction.

SOLUTION

1. Fraction to decimal: divide top by bottom, $3 \div 8 = 0.375$. *bar means ÷*
2. Decimal to fraction: 0.35 is "thirty-five hundredths", so write $\frac{35}{100}$. *35 over 100*
3. Simplify $\frac{35}{100}$ by dividing top and bottom by 5: $\frac{7}{20}$.

$\frac{3}{8} = 0.375$ **and** $0.35 = \frac{7}{20}$

Example 7

Compute $9.4 + 0.85 + 12.073$.

SOLUTION

1. Different lengths, so pad every number to **three** decimal places and align the points: 9.400, 0.850, 12.073. *pad to 3 places*
2. Add the thousandths: $0 + 0 + 3 = 3$. Add the hundredths: $0 + 5 + 7 = 12$, write 2 carry 1. *← carry the 1*
3. Tenths: $4 + 8 + 0 + 1 = 13$, write 3 carry 1; ones: $9 + 0 + 2 + 1 = 12$, write 2 carry 1; tens: $0 + 0 + 1 + 1 = 2$.
4. Bring the point straight down: 22.323.

$9.4 + 0.85 + 12.073 = 22.323$

Example 8

Compute $7.5 \div 0.4$, giving an exact decimal answer.

SOLUTION

1. Clear the point in the divisor: shift 0.4 one place right to 4, and the dividend $7.5 \to 75$. The problem is now $75 \div 4$. *×10 to both*
2. Long-divide: 4 into 7 goes 1 (remainder 3); 4 into 35 goes 8 (remainder 3). So far 18 remainder 3. *bring down digits*
3. The remainder is not 0, so add a decimal point and a zero: 4 into 30 goes 7 (remainder 2); another zero, 4 into 20 goes 5 (remainder 0). *annexe zeros*
4. Reading the quotient: 18.75.

$7.5 \div 0.4 = 18.75$

Example 9

A roll of wire is 2.6 m long and costs 1.35 dollars per metre. Find the total cost, rounded to the nearest cent.

SOLUTION

1. "Per metre" means multiply length by price: 2.6×1.35. Drop the points and multiply the whole numbers $26 \times 135 = 3510$. *drop the points first*
2. Count decimal places in the factors: 2.6 has 1 and 1.35 has 2, a total of 3. Place the point: $3510 \to 3.510$. *1 + 2 = 3 places*
3. Round to the nearest cent (two decimal places). The third decimal is 0, which is ≤ 4, so the hundredths digit stays. *0 → round down*
4. Drop the trailing digit: 3.51 dollars.

$2.6 \times 1.35 = 3.510 \approx 3.51$ **dollars**

Note: A fraction gives a **terminating** decimal exactly when, in lowest terms, its denominator has only 2s and 5s as prime factors (like $8 = 2^3$ or $20 = 2^2 \times 5$). Otherwise the decimal **repeats**, e.g. $\frac{1}{3} = 0.333\ldots$

decimal point tenths place value rounding terminating decimal

Mental-Math Shortcut

Use **front-end estimation** to get a fast, near-enough answer and to sanity-check your exact work: keep the leading (front) digits, round the rest, and combine.

For $4.82 + 6.13 + 2.95$, the front digits give $4 + 6 + 2 = 12$; the leftover tenths roughly add another ≈ 2, so the total is about 14 – and the exact answer 13.90 confirms it.

For a **product** like 19.7×4.1, round to friendly numbers: $20 \times 4 = 80$, so the true value (80.77) must land near 80 – instantly catching any misplaced decimal point. Whenever multiplication "loses" or "gains" a factor of ten, this estimate is what flags it.

To **multiply or divide by powers of ten**, just slide the point: $\times 10$ moves it one place right, $\div 100$ moves it two places left. So $3.7 \times 100 = 370$ and $42.5 \div 10 = 4.25$.

⚠ Common Mistakes

- Right-aligning the **digits** instead of the **decimal points** when adding, e.g. lining up 12.4 and 3.76 by their last digits. Always align the points (pad 12.4 to 12.40).
- Mis-counting decimal places in a product. 0.3×0.2 has $1 + 1 = 2$ decimal places, so it is 0.06, **not** 0.6.
- Shifting the point in only the divisor when dividing. If you move it in 0.9, you must move it the **same** number of places in the dividend too.
- Chain-rounding: rounding 3.146 to 3.15 then to 3.2. Always round in **one** step from the original digit – 3.146 to one decimal place is 3.1, because the deciding digit is 4.
- Assuming $0.1 + 0.2$ is "exactly" 0.3 on a computer. In binary floating-point it is not – see the Python box below.

Quiz

15.1.Q1. What is the value of the digit 7 in 3.275?

A. 7

B. 0.7

C. 0.07

D. 0.007

15.1.Q2. Compute $5.6 + 0.84$.

15.1.Q3. Compute $7 - 0.45$.

15.1.Q4. Compute 0.4×0.6.

A. 2.4

B. 0.24

C. 0.024

D. 0.0024

15.1.Q5. Compute $6.4 \div 0.8$.

A. 0.8

B. 8

C. 80

D. 0.08

15.1.Q6. Round 0.0639 to two decimal places.

15.1.Q7. Write $\frac{7}{20}$ as a decimal.

15.1.Q8. Write 0.125 as a fraction in lowest terms.

A. $\frac{1}{8}$

B. $\frac{125}{100}$

C. $\frac{1}{4}$

D. $\frac{12}{5}$

15.1.Q9. True or False: $0.7 = 0.70$.

A. True

B. False

🐍 From Concept to Code (Python)

Ordinary Python **floats** are binary fractions, so most decimals can only be stored *approximately*. The famous example is `0.1 + 0.2`. When you need exact base-ten arithmetic – money, especially – reach for **`decimal.Decimal`** instead.

```
print(0.1 + 0.2)              # binary float surprise
print(0.1 + 0.2 == 0.3)       # so this is False!
print(round(0.1 + 0.2, 2))    # round to tidy up for display
```

```
output
0.30000000000000004
False
0.3
```

```
from decimal import Decimal

# Build Decimals from strings to stay exact
a = Decimal("0.1") + Decimal("0.2")
print(a)
print(a == Decimal("0.3"))
print(Decimal("12.4") + Decimal("3.76"))
```

```
output
0.3
True
16.16
```

Try it:

1. Show the difference: print `0.1 + 0.2` as a float, then as `Decimal("0.1") + Decimal("0.2")`.
2. Use `round(x, 2)` to display 3.14159 to two decimal places, matching the chapter example.

Answer Key – Chapter 15 – Decimals

15.1.Q1. 0.07 – it sits in the hundredths place.

15.1.Q2. 6.44 – pad to $5.60 + 0.84$ and add.

15.1.Q3. 6.55 – write 7 as 7.00, then subtract.

15.1.Q4. 0.24 – $4 \times 6 = 24$ with 2 decimal places.

15.1.Q5. 8 – shift both points one place to get $64 \div 8 = 8$.

15.1.Q6. 0.06 – the deciding digit (3) is ≤ 4, so the hundredths stay.

15.1.Q7. 0.35 – $7 \div 20 = 0.35$ (or scale to $\frac{35}{100}$).

15.1.Q8. $\frac{1}{8}$ – $\frac{125}{1000}$ reduces by 125.

15.1.Q9. True – trailing zeros after the point add no value.

Chapter 16 · Percentages

How to convert between percentages, decimals and fractions; find a percentage of a number; apply a percentage increase or decrease; and reverse the process to find the original amount.

The Idea Behind It

A tricky "8% of 50" equals an easy "50% of 8": $a\%$ of $b = b\%$ of a. *but why?*

This swap looks like a coincidence, but it falls straight out of two plain facts. First, the word **percent** just means "divide by 100" (*per cent* = per hundred), and the word **"of"** means multiply:

$$\boxed{a\% = \frac{a}{100}, \qquad a\% \text{ of } b = \frac{a}{100} \times b}$$

Tidy that into a single fraction, where a and b sit side by side on top:

$$a\% \text{ of } b = \frac{a}{100} \times b = \frac{a \times b}{100}$$

Multiplication does not care about order ($a \times b = b \times a$), so the top is unchanged if we swap the letters, and we can read the very same fraction back as "$b\%$ of a":

$$\frac{a \times b}{100} = \frac{b \times a}{100} = \frac{b}{100} \times a = b\% \text{ of } a$$

$$\therefore \; a\% \text{ of } b = b\% \text{ of } a, \qquad 8\% \text{ of } 50 = 50\% \text{ of } 8 = 4$$

So whenever one side is awkward, flip the percent and the number: half of 8 is obviously 4. The trick is really just "$a \times b = b \times a$" wearing a percent sign. *flip to the easy half*

A **percentage** is simply a fraction whose denominator is fixed at 100 – the word *per cent* means "per hundred". Writing parts out of the same 100 every time makes amounts easy to compare: a test score of $\frac{17}{20}$ and one of $\frac{43}{50}$ are hard to rank at a glance, but as 85% and 86% the comparison is instant. Percentages are everywhere – discounts, interest, statistics – so fluency here pays off daily.

PERCENTAGE

$n\%$ means $\frac{n}{100}$ – that is, n parts out of every 100. So $37\% = \frac{37}{100} = 0.37$.

Converting between forms

Percentages, decimals and fractions are three costumes for the same number: - **Percent → decimal:** divide by 100 (move the point two places **left**). $45\% = 0.45$. - **Decimal → percent:** multiply by 100 (move the point two places **right**). $0.08 = 8\%$. - **Percent → fraction:** write it over 100 and simplify. $40\% = \frac{40}{100} = \frac{2}{5}$. - **Fraction → percent:** divide top by bottom, then multiply by 100. $\frac{3}{8} = 0.375 = 37.5\%$.

A percentage of a number

To find a percentage **of** an amount, turn the percent into a decimal (or fraction) and multiply. The word "of" signals multiplication.

$$n\% \text{ of } A = \frac{n}{100} \times A$$

Increase and decrease

To **increase** a quantity by $r\%$, you keep the original 100% and add $r\%$, so you multiply by the **multiplier** $\left(1 + \frac{r}{100}\right)$. To **decrease**, you keep $100\% - r\%$, multiplying by $\left(1 - \frac{r}{100}\right)$.

$$\text{new} = A\left(1 + \tfrac{r}{100}\right) \quad \text{or} \quad A\left(1 - \tfrac{r}{100}\right)$$

A 20% rise uses multiplier 1.20; a 20% fall uses 0.80. This single-multiplier idea also makes **reverse percentage** problems clean: if a price already *includes* a change, divide by the multiplier to undo it.

Worked examples

Example 1

Find 15% of 240.

SOLUTION

1. Convert the percent to a decimal: $15\% = 0.15$. ÷100 = move 2 left
2. The word "of" means multiply: $0.15 \times 240 = 36$.
3. Check with the building blocks: 10% of $240 = 24$ and 5% is half of that $= 12$, and $24 + 12 = 36$.

10% + 5%

36

Example 2

What percentage is 18 out of 40?

SOLUTION

1. Write "part out of whole" as a fraction: $\frac{18}{40}$. part / whole
2. Divide top by bottom: $18 \div 40 = 0.45$.
3. Multiply by 100 to make it a percent: $0.45 = 45\%$. ×100 = move 2 right

45%

Example 3

Increase 80 dollars by 25%.

SOLUTION

1. Keep the original 100% and add 25%: the multiplier is $1 + 0.25 = 1.25$. *100% + 25% = 1.25*
2. Multiply the original by it: $80 \times 1.25 = 100$.
3. Sanity check: 25% of 80 is 20, and $80 + 20 = 100$.

100 **dollars**

Example 4

A jacket costing 60 dollars is reduced by 30%. Find the sale price.

SOLUTION

1. A decrease keeps $100\% - 30\%$, so the multiplier is $1 - 0.30 = 0.70$. *you keep 70%*
2. Multiply the original by it: $60 \times 0.70 = 42$.
3. Check: 30% of 60 is 18, and $60 - 18 = 42$.

42 **dollars**

Example 5

After a 20% increase a salary is $36{,}000$. What was the original salary? (reverse percentage)

SOLUTION

1. The new amount is 120% of the original, i.e. the original was multiplied by 1.20. *forward multiplier*
2. To undo the increase, **divide** by that multiplier (do not subtract 20% of the new figure): $36000 \div 1.20$. *÷ to reverse*
3. Compute: $36000 \div 1.2 = 30000$.

$30{,}000$ **(and a check:** $30000 \times 1.2 = 36000$ ✓**)**

Example 6

A coat costs 63 dollars after a 10% discount. Find the original price.

SOLUTION

1. The sale price is 90% of the original, so it was multiplied by 0.90. *kept 90%*
2. Divide to reverse: $63 \div 0.90$. *undo with ÷*
3. Compute: $63 \div 0.9 = 70$.

70 **dollars**

Example 7

Find the simple interest on 500 dollars at 4% per year for 3 years.

SOLUTION

1. Simple interest is $\text{principal} \times \text{rate} \times \text{time}$, with the rate as a decimal: $500 \times 0.04 \times 3$. 4% → 0.04
2. One year of interest: $500 \times 0.04 = 20$.
3. Over three years: $20 \times 3 = 60$.

60 **dollars of interest (so the total owed is** 560**)**

Example 8

A share price rises from 40 dollars to 46 dollars. What is the percentage increase?

SOLUTION

1. Percentage change is $\dfrac{\text{change}}{\text{original}} \times 100$, always over the **original** value. The change is $46 - 40 = 6$.

   change = new − old
2. Divide by the original, not the new figure: $6 \div 40 = 0.15$. ÷ original = 40
3. Multiply by 100 to make it a percent: $0.15 = 15\%$. ×100 = move 2 right

15% **increase**

Example 9

A coat priced at 200 dollars is increased by 10%, then that new price is reduced by 10% in a sale. Find the final price.

SOLUTION

1. A rise of 10% uses multiplier 1.10; a fall of 10% uses multiplier 0.90. rise then fall
2. Apply them in turn to 200: $200 \times 1.10 = 220$, then $220 \times 0.90 = 198$.
3. Note it is **not** back to 200: combining the multipliers, $1.10 \times 0.90 = 0.99$, so the net effect is a 1% fall.

   rise + fall ≠ 0

198 **dollars (a net decrease of** 1%**)**

Example 10

Find the value of 1000 dollars after 2 years of **compound** interest at 5% per year.

SOLUTION

1. Compound interest applies the multiplier **each** year, on the running total. The yearly multiplier is $1 + 0.05 = 1.05$. 5% → ×1.05
2. After year one: $1000 \times 1.05 = 1050$. interest on 1000
3. After year two, apply the multiplier again to the **new** balance: $1050 \times 1.05 = 1102.50$.
   interest on 1050
4. Equivalently, multiply by 1.05 twice in one go: $1000 \times 1.05^2 = 1000 \times 1.1025 = 1102.50$.
   ×1.05 squared

1102.50 **dollars (compound interest of** 102.50**, more than simple interest of** 100**)**

Note: A 20% rise followed by a 20% fall does **not** return you to the start: $100 \to 120 \to 96$. The fall is taken from the *larger* amount, so percentage changes do not simply cancel. Multiply the multipliers ($1.2 \times 0.8 = 0.96$) to see the net effect.

percentage multiplier percentage increase reverse percentage principal

⚡ Mental-Math Shortcut

Build any percentage from two cheap blocks: 10% (divide by 10) and 1% (divide by 100).

Working out 35% of 80? Get $10\% = 8$, so $30\% = 24$; then 5% is half of 10%, i.e. 4. Add: $24 + 4 = 28$.

Other handy blocks: 50% is half, 25% is a quarter, and 5% is half of 10%. So $15\% = 10\% + 5\%$ and $20\% = 2 \times 10\%$.

A lovely shortcut from the symmetry $x\%$ of $y = y\%$ of x: a tricky "8% of 50" becomes the obvious "50% of 8" $= 4$. And "4% of 25" flips to "25% of 4" $= 1$.

⚠ Common Mistakes

- Forgetting to convert the percent before multiplying – computing 25×80 instead of 0.25×80 for "25% of 80". Always divide the percent by 100 first.
- Stopping at the *amount* of change instead of the new total. "80 increased by 25%" is 100, not the 20 of increase by itself.
- Doing reverse percentage by taking the percentage of the **new** figure. To undo a 20% rise you divide by 1.2; you do not subtract 20% of the new amount.
- Believing a 20% rise then a 20% fall cancels out. It gives 96% of the original, because the two changes act on different starting amounts.
- Mis-moving the decimal point: writing 0.5% as 0.5 instead of 0.005. Per cent means *two* places, so $0.5\% = 0.005$.

Quiz

16.1.Q1. Write 0.6 as a percentage.

A. 0.6%

B. 6%

C. 60%

D. 600%

16.1.Q2. Write $\frac{3}{5}$ as a percentage.

16.1.Q3. Find 30% of 90.

16.1.Q4. What percentage is 12 out of 48?

A. 4%

B. 12%

C. 25%

D. 36%

16.1.Q5. Increase 120 by 15%.

16.1.Q6. Decrease 50 by 40%.

16.1.Q7. After a 25% increase a price is 75. What was the original?

A. 50

B. 56.25

C. 60

D. 100

16.1.Q8. Simple interest on 200 at 5% for 2 years?

16.1.Q9. True or False: increasing by 50% then decreasing by 50% returns the original amount.

A. True

B. False

🐍 From Concept to Code (Python)

Percentages are easy to wrap in small **calculator functions**. Remember that "percent" is just "divide by 100", and Python's f-string format `{x:.0%}` prints a fraction *as* a percent for you.

```
def percent_of(p, amount):
    return p / 100 * amount

def apply_change(amount, p):     # +p for increase, -p for decrease
    return amount * (1 + p / 100)

print(percent_of(15, 240))       # 15% of 240
print(apply_change(80, 25))      # 80 increased by 25%
print(apply_change(60, -30))     # 60 decreased by 30%
```

```
output
36.0
100.0
42.0
```

```
score = 18 / 40
print(f"{score:.0%}")            # format a fraction as a percent
print(f"{0.375:.1%}")           # one decimal place
```

```
output
45%
37.5%
```

Try it:

1. Write a `reverse_increase(new, p)` function that returns `new / (1 + p/100)`; test it on 36000 after a 20% rise.
2. Use an f-string to print $\frac{43}{50}$ as a percentage.

Answer Key – Chapter 16 – Percentages

16.1.Q1. 60% – multiply by 100.

16.1.Q2. 60% – $3 \div 5 = 0.6 = 60\%$.

16.1.Q3. 27 – $0.30 \times 90 = 27$ (or $10\% = 9$, times 3).

16.1.Q4. 25% – $\frac{12}{48} = \frac{1}{4} = 25\%$.

16.1.Q5. 138 – multiplier 1.15: $120 \times 1.15 = 138$.

16.1.Q6. 30 – multiplier 0.60: $50 \times 0.6 = 30$.

16.1.Q7. 60 – divide by 1.25: $75 \div 1.25 = 60$.

16.1.Q8. 20 – $200 \times 0.05 \times 2 = 20$.

16.1.Q9. False – $1.5 \times 0.5 = 0.75$, i.e. 75% of the original.



PART V

Introduction to Logic

The grammar of correct reasoning – and the bridge to programming. Statements, connectives, truth tables, conditionals, sets, Boolean algebra, and how to tell a valid argument from a fallacy.

Chapter 17 – Statements & Propositions · Chapter 18 – Logical Connectives: AND, OR, NOT · Chapter 19 – Truth Tables · Chapter 20 – Conditionals & Implication · Chapter 21 – Equivalence & De Morgan's Laws · Chapter 22 – Quantifiers: For All & There Exists · Chapter 23 – Sets & Venn Diagrams · Chapter 24 – Boolean Algebra · Chapter 25 – Logical Reasoning & Validity

Chapter 17 · Statements & Propositions

How to decide whether a sentence is a logical statement, give it a truth value, name it with a letter, and negate it.

The Idea Behind It

A proposition must be exactly one of true or false, which is why *"this sentence is false"* is not a proposition. *but why?*

A statement is a claim that the world has *settled* one way: either it holds or it does not, with no third option and no both-at-once. That single demand, called **bivalence**, is the entrance test for being a proposition. The cleanest way to see a sentence **fail** the test is to try both labels and watch them collapse, so take the liar sentence L: *"this sentence is false."* There are only two truth values to try, so check each:

Case T. Suppose L is T. Then what L asserts is so, and it asserts that it is false, so L is F. We assumed T and derived F: a contradiction.

Case F. Suppose L is F. Then what L asserts is *not* so; it asserts it is false, so it is *not* false, i.e. L is T. We assumed F and derived T: again a contradiction. Laid side by side:

Assume L is	What L claims forces	Result
T	F	contradiction
F	T	contradiction

Both of the only two options self-destruct, so no consistent truth value exists for L.

$$\therefore\ L \text{ is } \text{not a proposition}$$

The test cuts both ways: questions and commands assert nothing, so they get no value either, while the liar asserts something that **cannot** hold a single value. A proposition is exactly a sentence the true/false test accepts. *settled one way, no paradox*

Logic is the study of correct reasoning, and it starts with a surprisingly simple question: *is this sentence true or false?* Mathematics deals only with sentences that have a definite answer. "Paris is the capital of France" is true. "7 is an even number" is false. Sentences like these are the raw material of all logic, so we give them a special name.

STATEMENT (PROPOSITION)

A **statement** (or **proposition**) is a declarative sentence that is either **true** or **false**, but not both. The label *true* or *false* it carries is called its **truth value**.

The two key words are *declarative* (it asserts something) and *definite* (there is a fact of the matter, even if you personally don't know it). A sentence does **not** have to be true to be a statement – it just has to be the kind of sentence that *can* be true or false.

What is NOT a statement

Many perfectly good English sentences are not statements, because they cannot be assigned a single truth value:

- **Questions** – "What time is it?" asks something; it is neither true nor false.
- **Commands / requests** – "Close the door." tells you to do something.
- **Exclamations** – "What a great film!"
- **Opinions & tastes** – "Chocolate is the best flavour." is a matter of preference, not fact.
- **Open sentences** with an unknown – "$x > 5$" is true for some x and false for others, so by itself it has no fixed truth value until x is specified.

The test is mechanical: *could a reasonable person, given enough information, label this sentence definitively true or definitively false?* If yes, it is a statement.

Naming statements with letters

Just as algebra uses x and y for numbers, logic uses lower-case letters – most often p, q, r – to stand for whole statements. This lets us study the *shape* of an argument without rewriting the English each time. We write, for example,

$$p : \text{"It is raining."} \qquad q : \text{"The ground is wet."}$$

Once p names a statement, p is either true (T) or false (F). We will usually abbreviate the truth values to the single letters **T** and **F**.

Negation

The simplest thing you can do to a statement is **deny** it. The negation of p is the statement "it is not the case that p", written $\neg p$ and read *"not p"*.

> **NEGATION**
>
> The **negation** of a statement p, written $\neg p$, is the statement that is **true exactly when p is false**, and false exactly when p is true. Negating simply flips the truth value.

If p is "The door is open," then $\neg p$ is "The door is **not** open," i.e. "The door is closed." Negating twice returns you to where you started: $\neg(\neg p)$ means the same as p – this is the **double-negative** rule, "it is not the case that the door is not open" just says the door is open.

Worked examples

Example 1

Decide whether each sentence is a statement. If it is, give its truth value. (a) "$5 + 7 = 12$." (b) "Is it cold outside?" (c) "Please sit down." (d) "The Moon is made of cheese."

SOLUTION

1. (a) Declarative and definite: $5 + 7$ really does equal 12, so it is true. **Statement, T.** *checkable fact*
2. (b) A question – it asserts nothing, so no truth value. **Not a statement.**
3. (c) A command – it tells you to act, it does not claim anything. **Not a statement.**
4. (d) Declarative and definite (we can check it against reality). It happens to be false, but it is still a statement. **Statement, F.** *false still counts*

(a) statement, T · (b) not a statement · (c) not a statement · (d) statement, F

Example 2

Is "$x + 2 = 9$" a statement?

SOLUTION

1. It looks like a fact, but its truth depends on the unknown x. *value not fixed*
2. If $x = 7$ it is true; if $x = 0$ it is false. There is no single truth value.
3. A sentence containing a free unknown like this is called an **open sentence**, not a statement, until x is given a value.

No – it is an open sentence (true only when $x = 7$).

Example 3

Write the negation of each in plain English. (a) p: "It is raining." (b) q: "9 is a prime number." (c) r: "Every student passed."

SOLUTION

1. (a) $\neg p$: "It is **not** raining." *just insert "not"*
2. (b) $\neg q$: "9 is **not** a prime number." (Here q is false, so $\neg q$ is true – indeed $9 = 3 \times 3$.) *flip the value*
3. (c) The careful negation of "every student passed" is "**not** every student passed," i.e. "**at least one** student did not pass." Beware: it is *not* "every student failed." *not all ≠ none*

(a) It is not raining. (b) 9 is not prime. (c) At least one student did not pass.

Example 4

Statement p is true. What is the truth value of $\neg p$? Of $\neg(\neg p)$?

SOLUTION

1. Negation flips the value, so if p is T then $\neg p$ is F. *flip once*
2. Negating again flips back: $\neg(\neg p)$ is T. *flip twice = back*
3. This confirms the double-negative rule: $\neg(\neg p)$ always has the same truth value as p.

$\neg p$ is F; $\neg(\neg p)$ is T.

Example 5

Classify and, where possible, give the truth value: "This sentence is false."

SOLUTION

1. It looks declarative, so we try to assign a truth value.
2. If it were true, then what it says holds – so it is false. Contradiction. *T forces F*
3. If it were false, then what it says fails – so it is true. Contradiction again. *F forces T*
4. No consistent truth value exists. This famous **liar paradox** is therefore *not* a proper statement.

Not a statement – it cannot consistently be true or false (a paradox).

Example 6

For each, decide statement or not; if a statement, give its truth value. (a) "7 is greater than 10." (b) "Some prime numbers are even." (c) "$2x = 10$." (d) "Tidy your room!"

SOLUTION

1. (a) Declarative and definite: $7 > 10$ is plainly false, but it is still a **statement, F**. *false still counts*
2. (b) "Some prime numbers are even" makes a definite claim; it is true because 2 is an even prime. **Statement, T.** *one example suffices*
3. (c) Contains a free unknown x, so its truth depends on the value – an **open sentence**, not a statement.
4. (d) A command; it asserts nothing. **Not a statement.**

(a) statement, F · (b) statement, T · (c) open sentence · (d) not a statement

Example 7

Write the careful negation of each. (a) p: "All cats are black." (b) q: "Some doors are open." (c) r: "No student is late."

SOLUTION

1. (a) The negation of "**all** are" is "**not all** are," i.e. "at least one cat is not black." It is *not* "no cat is black."
   not all ≠ none
2. (b) The negation of "**some** are" is "**none** are," i.e. "no door is open" (every door is closed).
   some → none
3. (c) "No student is late" already means "every student is not late"; its negation is "at least one student is late." *no → at least one*

(a) At least one cat is not black. (b) No door is open. (c) At least one student is late.

Example 8

Statement p is false. Work through the truth values of $\neg p$, $\neg(\neg p)$ and $\neg(\neg(\neg p))$.

SOLUTION

1. Start from p being F. Each $\neg$ flips the value once, so build the chain step by step. *flip each time*
2. The small table records the flips: an **odd** number of negations reverses p, an **even** number returns it.
   odd flips, even keeps
3. Reading the last row, $\neg(\neg(\neg p))$ has the same value as a single $\neg p$, namely T.

$\neg p$ is T; $\neg(\neg p)$ is F; $\neg(\neg(\neg p))$ is T.

expression	value
p	F
$\neg p$	T
$\neg(\neg p)$	F
$\neg(\neg(\neg p))$	T

Note: "Not true" and "false" mean the same thing for statements, because a statement has exactly two possible values. That two-value assumption – every statement is either true or false, with no third option – is called the **law of the excluded middle**, and the whole of this strand relies on it.

statement truth value proposition open sentence negation

⚠ Common Mistakes

- Thinking a sentence must be **true** to count as a statement. False sentences like "$2 + 2 = 5$" are perfectly good statements – they just have truth value F.
- Treating **questions, commands, or opinions** as statements. "Shut the window!" and "Pizza is delicious" carry no objective truth value.
- Calling an **open sentence** like "$x > 3$" a statement. Until x is given a value it is neither true nor false.
- Negating "all/every" incorrectly: the negation of "all students passed" is "**not all** passed" (at least one failed), **not** "all students failed."
- Forgetting the **double-negative** rule: $\neg(\neg p)$ is just p, not a new, stronger statement.

📝 Quiz

17.1.Q1. Which of these is a statement (proposition)?

A. "Close the door."

B. "Is 7 prime?"

C. "12 is divisible by 4."

D. "What a day!"

17.1.Q2. True or False: a sentence must be true in order to be a statement.

A. True

B. False

17.1.Q3. What is the truth value of the statement "$3 \times 4 = 11$"?

17.1.Q4. Is "$x - 1 = 4$" a statement?

A. Yes, it is true

B. Yes, it is false

C. No, it is an open sentence

D. It is a command

17.1.Q5. If p is the statement "The light is on," write $\neg p$ in plain English.

17.1.Q6. Statement q is false. What are the truth values of $\neg q$ and $\neg(\neg q)$?

17.1.Q7. True or False: "This sentence is false" is a statement with truth value T.

A. True

B. False

17.1.Q8. Write the correct negation of "Every door is locked."

🐍 From Concept to Code (Python)

Python has a dedicated type for truth values: **`bool`**, whose only values are `True` and `False` (note the capital letters). A Python comparison *is* a statement – it evaluates to one of those two values. The `not` keyword is exactly negation ¬: it flips `True` to `False` and back.

```
p = (3 * 4 == 12)      # a statement: "3*4 equals 12"
q = (9 < 5)             # a false statement
print(p, q)
print(not p)           # negation flips the value
print(not (not q))     # double negation returns q
```

```
output
True False
False
False
```

Try it:

1. Store the statement "7 is even" as `p = (7 % 2 == 0)`, then print `p` and `not p`.
2. Predict the output of `print(not True, not False)`, then run it to check.

Answer Key – Chapter 17 – Statements & Propositions

17.1.Q1. "12 is divisible by 4." – it is declarative and definite (true). The others are a command, a question, and an exclamation.

17.1.Q2. False – a statement only needs a definite truth value; false sentences still qualify.

17.1.Q3. False – $3 \times 4 = 12$, so the statement is F (but it is still a valid statement).

17.1.Q4. No – it is an open sentence; true only when $x = 5$, false otherwise, so it has no fixed truth value.

17.1.Q5. "The light is not on" (equivalently, "The light is off").

17.1.Q6. $\neg q$ is T (negation flips); $\neg(\neg q)$ is F again (double negation returns the original).

17.1.Q7. False – it is a paradox and cannot be consistently assigned any truth value, so it is not a statement at all.

17.1.Q8. "Not every door is locked," i.e. "At least one door is unlocked." (Not "Every door is unlocked.")

Chapter 18 · Logical Connectives: AND, OR, NOT

How to combine statements with the connectives AND ($\wedge$), OR ($\vee$) and NOT ($\neg$), and why logical "or" differs from everyday "or".

The Idea Behind It

Logical *or* is inclusive: $p \vee q$ is false only when both parts are false, so "both true" still makes it true.

but why?

A connective is not an opinion; it *is* its truth table. To pin down what $\vee$ means we simply decide a value for every way p and q can turn out, and the guiding idea is the plain meaning of *or*: the claim "p or q" is satisfied the moment **at least one** part holds. The only way to leave "at least one" unsatisfied is for every part to fail. Two statements give $2^2 = 4$ rows, so write them all and apply that rule:

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

Read the result column down: it is T on the top three rows (where one part, or both, are true) and F on the single all-false bottom row.

$$\therefore\ p \vee q \text{ is } \text{false only when both are false}$$

The top row is the whole story: in logic $\mathsf{T} \vee \mathsf{T} = \mathsf{T}$, so "both" never breaks an *or*, it satisfies it. Everyday speech often means the **exclusive** "one or the other but not both", but the default $\vee$ of mathematics is this inclusive one. *both true still counts*

A single statement can only say so much. The power of logic comes from **joining** statements into bigger ones. The little words that do the joining – *and*, *or*, *not* – are called **logical connectives**, and each has a precise mathematical meaning that never changes, no matter what the statements are about.

Throughout this chapter let p and q stand for any two statements. A statement built from smaller ones using connectives is called a **compound statement**; the smallest, unbreakable pieces are **simple** (or atomic) statements.

Conjunction – AND

CONJUNCTION ($\LAND$)

The **conjunction** of p and q, written $p \wedge q$ and read *"p and q"*, is **true only when both p and q are true**. If either part is false, the whole conjunction is false.

Think of *and* as demanding **everything at once**. "It is raining $\wedge$ I have an umbrella" is true only on the rainy day when you also happen to be carrying the umbrella. The order does not matter: $p \wedge q$ means the same as $q \wedge p$.

Disjunction – OR (inclusive)

> **DISJUNCTION ($\LOR$)**
>
> The **disjunction** of p and q, written $p \vee q$ and read *"p or q"*, is **true when at least one of p, q is true** – and so it is false only when **both** are false.

This is the crucial point that trips everyone up. In logic, *or* is **inclusive**: "p or q" allows the possibility that **both** are true. "You may have tea $\vee$ coffee" in logic means you could have tea, or coffee, **or both**. Mathematicians always mean inclusive *or* unless they say otherwise.

> **Note:** Everyday English often uses *or* in the **exclusive** sense – "one or the other, but not both." A menu that says "soup or salad" usually means *exactly one*. That everyday meaning is the **exclusive or** ($\oplus$), and it is **not** the default $\vee$ of logic. When this book writes $p \vee q$, "both true" makes it true.

Negation – NOT

We met **negation** $\neg p$ ("not p") last chapter: it simply flips a truth value, turning T into F and F into T. Negation acts on a **single** statement, whereas $\wedge$ and $\vee$ join **two**. We can of course negate a compound statement: $\neg(p \wedge q)$ denies that both hold.

A note on grouping

As with arithmetic, **brackets decide what attaches to what**. The convention is that $\neg$ binds **tightest** (it grabs only the statement immediately after it), then $\wedge$, then $\vee$ – but when in doubt, *use brackets*. So $\neg p \wedge q$ means $(\neg p) \wedge q$, whereas $\neg(p \wedge q)$ negates the whole conjunction. These two say very different things.

$$(\neg p) \wedge q \quad \text{is NOT the same as} \quad \neg(p \wedge q)$$

Worked examples

Example 1

Let p: "$5 > 3$" (true) and q: "5 is even" (false). Find the truth value of $p \wedge q$ and of $p \vee q$.

> **SOLUTION**
>
> 1. p is T and q is F.
> 2. Conjunction needs **both** true. Since q is F, $p \wedge q$ is F. *one F sinks AND*
> 3. Disjunction needs **at least one** true. Since p is T, $p \vee q$ is T. *one T floats OR*
>
> $p \wedge q$ **is F**; $p \vee q$ **is T**.

Example 2

A shop sign reads: "Open if it is a weekday $\vee$ it is a holiday." On a Saturday (weekday: F, holiday: F), is the shop open? On a public holiday that falls on a weekday (weekday: T, holiday: T)?

SOLUTION

1. Saturday: both parts are F, and a disjunction is false only when **both** parts are false – so $\mathrm{F} \vee \mathrm{F}$ is F, the shop is **closed**.
2. Holiday on a weekday: both parts are T. Inclusive $\vee$ is **true** when both are true, so $\mathrm{T} \vee \mathrm{T}$ is T – the shop is **open**. both T → still T
3. This shows inclusive *or*: "both" does not break the disjunction, it satisfies it.

Saturday: closed (F$\vee$F). Holiday-weekday: open (T$\vee$T).

Example 3

Translate into symbols, using r: "it is raining" and w: "the ground is wet." (a) "It is raining and the ground is wet." (b) "It is not raining." (c) "It is raining or the ground is not wet."

SOLUTION

1. (a) Both joined by *and*: $r \wedge w$. and → $\wedge$
2. (b) A single negation: $\neg r$. not → $\neg$
3. (c) "r" or "(not w)", joined by inclusive *or*: $r \vee (\neg w)$. or → $\vee$

(a) $r \wedge w$ **(b)** $\neg r$ **(c)** $r \vee \neg w$

Example 4

With p true and q true, evaluate $(\neg p) \wedge q$ and $\neg(p \wedge q)$ to show they differ.

SOLUTION

1. First $(\neg p) \wedge q$: negate p to get $\neg p = \mathrm{F}$, then $\mathrm{F} \wedge \mathrm{T} = \mathrm{F}$ (q is T). ¬ grabs only p
2. Now $\neg(p \wedge q)$: inside, $p \wedge q$ is $\mathrm{T} \wedge \mathrm{T} = \mathrm{T}$; negating gives F. ¬ wraps the whole bracket
3. They happen to agree here (both F), so try p true, q **false**: $(\neg p) \wedge q = \mathrm{F} \wedge \mathrm{F} = \mathrm{F}$, but $\neg(p \wedge q) = \neg(\mathrm{F}) = \mathrm{T}$. Now they disagree – proving the bracket matters.

They are different statements: at p =T, q =F they give F and T respectively.

Example 5

Is the everyday sentence "Tea or coffee?" usually inclusive or exclusive *or*? How would you write a strict "exactly one of p, q" using $\wedge, \vee, \neg$?

SOLUTION

1. A waiter offering "tea or coffee" normally expects you to pick **one**, not both – that is **exclusive** *or*.

   everyday = one only

2. "Exactly one" means: at least one is true, **and** not both are true.
3. In symbols: $(p \vee q) \wedge \neg(p \wedge q)$ – the $(p \vee q)$ supplies "at least one," and $\neg(p \wedge q)$ rules out "both."

   at least one, but not both

Everyday: exclusive. Exactly-one $= (p \vee q) \wedge \neg(p \wedge q)$**.**

Example 6

Let p be true, q false, r true. Evaluate $(p \wedge \neg q) \vee r$ step by step.

SOLUTION

1. Work the innermost piece first: negate q. Since q is F, $\neg q$ is T. *flip q*
2. Now the bracket: $p \wedge \neg q = \mathrm{T} \wedge \mathrm{T} = \mathrm{T}$ (both parts true). *AND needs both*
3. Finally the outer $\vee$: $\mathrm{T} \vee r = \mathrm{T} \vee \mathrm{T} = \mathrm{T}$. A disjunction is true once either side is. *one T floats OR*

$(p \wedge \neg q) \vee r = T$**.**

Example 7

Build the truth table for $(\neg p) \vee q$, working a helper column first.

SOLUTION

1. Two letters $\Rightarrow 2^2 = 4$ rows. List p, q in standard order and add a helper column for $\neg p$ (flip the p column).

   build inside-out

2. Form the disjunction $(\neg p) \vee q$: it is T whenever $\neg p$ is T **or** q is T. *OR: at least one*
3. Reading the final column, it is false **only** on the row $p =$ T, $q =$ F, where $\neg p$ is F and q is F.

False only on the row p= T, q= F; true on the other three.

p	q	$\neg p$	$(\neg p) \vee q$
T	T	F	T
T	F	F	F
F	T	T	T
F	F	T	T

Example 8

Translate "It is sunny, and it is not both windy and cold" into symbols, using s (sunny), w (windy), c (cold).

SOLUTION

1. Split on the main *and*: the sentence is "s" and "(not both windy and cold)". outer connective = ∧
2. "Both windy and cold" is the conjunction $w \wedge c$; "not both" negates it: $\neg(w \wedge c)$. not both → ¬(…)
3. Join the two parts with the main ∧: $s \wedge \neg(w \wedge c)$. The brackets keep the ¬ on the whole "$w \wedge c$".

brackets matter

$s \wedge \neg(w \wedge c)$.

Note: A quick way to remember the two main connectives: ∧ (AND) is **strict** – it fails the moment *anything* is false. ∨ (OR) is **generous** – it succeeds the moment *anything* is true, and only fails when everything is false.

connective conjunction disjunction inclusive or exclusive or compound statement

⚠ Common Mistakes

- Reading logical "or" as **exclusive**. In mathematics $p \vee q$ is **inclusive** – it is true even when both p and q are true; it is false only when both are false.
- Thinking a conjunction $p \wedge q$ can be true when only one part is true. It needs **both** parts true; a single F makes the whole thing F.
- Dropping brackets and confusing $\neg p \wedge q$ with $\neg(p \wedge q)$. Negation grabs only the statement right after it unless brackets say otherwise.
- Believing a disjunction is false when one part is false. $p \vee q$ only becomes false when **every** part is false.
- Assuming order matters: $p \wedge q$ and $q \wedge p$ (and likewise for ∨) always have the same truth value.

Quiz

18.1.Q1. For $p \wedge q$ to be **true**, how many of p, q must be true?

A. At least one

B. Exactly one

C. Both

D. Neither

18.1.Q2. True or False: in logic, $p \vee q$ is **false** when both p and q are true.

A. True

B. False

18.1.Q3. Let p: "$4 > 2$" (T) and q: "4 is odd" (F). What is $p \wedge q$?

18.1.Q4. With the same p, q as above, what is $p \vee q$?

18.1.Q5. Translate "It is cold and it is not snowing" using c (cold) and s (snowing).

18.1.Q6. Which everyday phrase usually means **exclusive** or?

A. "Bring a coat or an umbrella (you may want both)"

B. "You can pay by card or cash" (pick one)

C. "$x > 0$ or $x < 0$"

D. "Tea contains caffeine or theine"

18.1.Q7. True or False: $\neg p \wedge q$ means the same as $\neg(p \wedge q)$.

A. True

B. False

18.1.Q8. Write "exactly one of p, q is true" using $\wedge, \vee, \neg$.

🐍 From Concept to Code (Python)

Python's keywords **`and`**, **`or`**, **`not`** are exactly the connectives $\wedge$, $\vee$, $\neg$. `and` is true only when both sides are true; `or` is **inclusive** (true when at least one side is true); `not` flips a value. They follow the same binding rules – `not` first, then `and`, then `or` – and you can use parentheses to group, just like in logic.

```
p, q = True, False
print(p and q)        # conjunction: both needed
print(p or q)         # inclusive disjunction
print(True or True)   # inclusive: both true -> still True
print(not p and q)    # binds as (not p) and q
```

```
output
False
True
True
False
```

Try it:

1. Print the truth value of `(5 > 3) and (4 % 2 == 0)`, then of the same with `or`.
2. Write a Python expression for "exactly one of `p`, `q` is true" using `and`, `or`, `not`, and test it on all four combinations of `True/False`.

Answer Key – Chapter 18 – Logical Connectives: AND, OR, NOT

18.1.Q1. Both – conjunction is true only when every part is true.

18.1.Q2. False – inclusive $\vee$ is **true** when both are true; it is false only when both are false.

18.1.Q3. False – conjunction needs both true, and q is false.

18.1.Q4. True – disjunction needs at least one true, and p is true.

18.1.Q5. $c \wedge \neg s$ – conjunction of c with the negation of s.

18.1.Q6. "Pay by card or cash" – you choose exactly one method, the exclusive sense.

18.1.Q7. False – $\neg$ binds tighter, so $\neg p \wedge q$ is $(\neg p) \wedge q$, which differs from negating the whole conjunction (e.g. at p =T, q=F they give F and T).

18.1.Q8. $(p \vee q) \wedge \neg(p \wedge q)$ – at least one true, but not both.

Chapter 19 · Truth Tables

How to build a truth table with all 2^n rows and use it to find the truth value of any compound statement.

The Idea Behind It

A truth table for n different statements needs exactly 2^n rows. *but why?*

A row of a truth table is one complete way the world could be: a choice of T or F for each simple statement. To miss none, you must count those choices. Each statement is **independently** either true or false, that is 2 options, and the statements do not constrain one another, so the choices **multiply** (the product rule of counting):

$$\underbrace{2 \times 2 \times \cdots \times 2}_{n \text{ statements}} = 2^n$$

The same fact seen as growth: adding **one** more statement splits every existing row into two (the new letter T, then F), so the row count **doubles** each time. Track it from the start:

statements n	rows	as a power
1	2	2^1
2	4	2^2
3	8	2^3
n	2^n	2^n

$$\therefore\ n \text{ statements need } 2^n \text{ rows}$$

So 3 letters need 8 rows and never 6: a missing row is a hidden case the statement might behave differently in. The doubling is also why tables grow alarmingly fast, 10 statements already demand 1024 rows.

each new letter doubles it

How can you be *certain* what a compound statement does in every situation? You list **every** possibility. A **truth table** is an organised grid that shows the truth value of a statement for each combination of truth values of its parts. It is the single most useful tool in logic: build the table, and the answer to "is this ever false?" or "do these two mean the same?" is right there in front of you.

TRUTH TABLE

A **truth table** lists, in its rows, **every possible combination** of truth values for the simple statements involved, and shows in further columns the resulting value of each compound statement. A statement built from n different simple statements needs 2^n rows.

Why 2^n? Each simple statement can be either **T** or **F** – that is 2 choices. With n independent statements the choices multiply: $2 \times 2 \times \cdots \times 2 = 2^n$. So 1 statement needs 2 rows, 2 statements need 4, 3 statements need 8, and so on.

Listing the rows in standard order

To make sure you miss nothing, fill the input columns in a fixed pattern. For two statements p, q, the standard order is TT, TF, FT, FF – the left column halves (T then F), the right column alternates:

p	q
T	T
T	F
F	T
F	F

The three basic tables

Everything is built from these. **Negation** flips the value:

p	$\neg p$
T	F
F	T

Conjunction $p \wedge q$ – true only on the all-true row:

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

Disjunction $p \vee q$ – false only on the all-false row (note the inclusive top row is **T**):

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

Building a compound table, column by column

For a bigger expression, **add one column per sub-step**, working from the inside out (just like order of operations), and finish with the whole expression. Each new column uses only columns already to its left. The final column is your answer.

Worked examples

Example 1

Build the truth table for $\neg p \vee q$.

SOLUTION

1. Two statements $\Rightarrow 2^2 = 4$ rows. List p, q in standard order.
2. Add a helper column for $\neg p$ (flip the p column). *build inside-out*
3. Form $\neg p \vee q$: it is T whenever $\neg p$ is T **or** q is T (false only on the row T, F). *rightmost col = answer*

p	q	$\neg p$	$\neg p \vee q$
T	T	F	T
T	F	F	F
F	T	T	T
F	F	T	T

Example 2

Build the truth table for $p \wedge \neg q$ and say on which row(s) it is true.

SOLUTION

1. $2^2 = 4$ rows. Add a helper column $\neg q$ (flip q). *flip q first*
2. The conjunction $p \wedge \neg q$ is true only where **both** p is T and $\neg q$ is T. *AND needs both*
3. That happens only on the row $p =$ T, $q =$ F (so $\neg q$ is T there).

True only on the row p=T, q=F.

p	q	$\neg q$	$p \wedge \neg q$
T	T	F	F
T	F	T	T
F	T	F	F
F	F	T	F

Example 3

Build the truth table for the compound statement $(p \vee q) \wedge \neg p$.

SOLUTION

1. 4 rows. Build inside-out: first the helper column $p \vee q$, then $\neg p$. *two helpers first*
2. Finally combine them with $\wedge$: the result $(p \vee q) \wedge \neg p$ is T only where **both** $(p \vee q)$ and $\neg p$ are T.
3. Reading the last column, it is true only when $p =$ F and $q =$ T – i.e. exactly when q is true but p is not.

True only on the row p= F, q= T.

p	q	$p \vee q$	$\neg p$	$(p \vee q) \wedge \neg p$
T	T	T	F	F
T	F	T	F	F
F	T	T	T	T
F	F	F	T	F

Example 4

A statement uses three letters: p, q, r. How many rows are needed, and what is the standard ordering of the input columns?

SOLUTION

1. Three simple statements $\Rightarrow 2^3 = 8$ rows. *each letter doubles the rows*
2. Standard order: the first column is T for the top half (4 rows) then F; the middle alternates in blocks of two; the last alternates every row. *halve, then halve again*
3. The table below shows the full 8-row input block.

8 rows.

p	q	r
T	T	T
T	T	F
T	F	T
T	F	F
F	T	T
F	T	F
F	F	T
F	F	F

Example 5

Build the truth table for the three-variable statement $p \wedge (q \vee r)$.

SOLUTION

1. $2^3 = 8$ rows. Build the helper column $q \vee r$ first (false only when both q, r are F).

   innermost bracket first

2. Then form $p \wedge (q \vee r)$: T only where p is T **and** $q \vee r$ is T. AND needs both
3. So every row with $p =$ F is automatically F, and among the $p =$ T rows it follows $q \vee r$.

See the final column below.

p	q	r	$q \vee r$	$p \wedge (q \vee r)$
T	T	T	T	T
T	T	F	T	T
T	F	T	T	T
T	F	F	F	F
F	T	T	T	F
F	T	F	T	F
F	F	T	T	F
F	F	F	F	F

Example 6

Build the truth table for $\neg p \vee \neg q$ and identify the row(s) where it is false.

SOLUTION

1. $2^2 = 4$ rows. Add two helper columns, $\neg p$ and $\neg q$ (flip each input column). flip both inputs
2. Form $\neg p \vee \neg q$: it is T whenever **at least one** of $\neg p, \neg q$ is T. OR: at least one
3. It is false only where both $\neg p$ and $\neg q$ are F – that is the single row $p =$ T, $q =$ T. false on TT only

False only on the row p=T, q=T.

p	q	$\neg p$	$\neg q$	$\neg p \vee \neg q$
T	T	F	F	F
T	F	F	T	T
F	T	T	F	T
F	F	T	T	T

Example 7

Build the truth table for $(p \wedge q) \vee (\neg p \wedge r)$.

SOLUTION

1. $2^3 = 8$ rows. Build two helper columns from the inside out: $p \wedge q$, and $\neg p \wedge r$ (which needs $\neg p$ first).

   two brackets first

2. Combine the two helpers with the outer $\vee$: $(p \wedge q) \vee (\neg p \wedge r)$ is T where **either** bracket is T.

   OR joins them

3. Reading the final column, it is true on the rows where p, q are both T, **or** where p is F while r is T.

True on rows TTT, TTF, FTT, FFT (where $p \wedge q$ holds, or $\neg p \wedge r$ holds).

p	q	r	$p \wedge q$	$\neg p \wedge r$	$(p \wedge q) \vee (\neg p \wedge r)$
T	T	T	T	F	T
T	T	F	T	F	T
T	F	T	F	F	F
T	F	F	F	F	F
F	T	T	F	T	T
F	T	F	F	F	F
F	F	T	F	T	T
F	F	F	F	F	F

Example 8

Show by truth table that $p \vee \neg p$ is a tautology (true on every row).

SOLUTION

1. Just one letter $\Rightarrow 2^1 = 2$ rows. Add a helper column for $\neg p$. *only 2 rows*
2. Form $p \vee \neg p$: on each row, exactly one of p, $\neg p$ is T, so the disjunction is T. *one side always T*
3. The final column is T on **both** rows – an all-T column, so the statement is a tautology.

   all T → tautology

Both rows give T, so $p \vee \neg p$ is a tautology (the "law of the excluded middle").

p	$\neg p$	$p \vee \neg p$
T	F	T
F	T	T

Note: A column that comes out **all T** describes a statement that is always true – a **tautology**. A column that is **all F** is a **contradiction** (always false). We will use both ideas in the next two chapters to test arguments and prove equivalences.

Tip: To avoid mistakes, build truth tables the way you evaluate arithmetic: **innermost brackets first**, one new column per operation, never skipping a step. Keep the simple-statement columns in the standard T/F order every time, so a half-finished table is easy to check.

truth table row count 2^n tautology contradiction

⚠ Common Mistakes

- Writing too few rows. With n simple statements you need 2^n rows – 3 statements need 8, not 6. A missing row can hide a case where the statement behaves differently.
- Filling the inputs haphazardly so some combination is repeated or omitted. Always use the standard order (left column half-T-half-F, each next column alternating twice as fast).
- Putting **F** in the top row of a disjunction. $p \vee q$ is inclusive, so the TT row is **T**.
- Skipping helper columns and trying to evaluate a big expression in one go. Build inside-out, one operation per column.
- Reading the wrong column as the answer – the **final** (rightmost, whole-expression) column is the result; the helpers are just scaffolding.

Quiz

19.1.Q1. How many rows does a truth table for a statement in p, q, r, s (four letters) have?

A. 8

B. 12

C. 16

D. 4

19.1.Q2. In the truth table for $p \vee q$, what is the value on the row $p =$T, $q =$T?

A. T

B. F

19.1.Q3. On which single row is $p \wedge q$ true?

19.1.Q4. True or False: a statement whose final column is all T is called a contradiction.

A. True

B. False

19.1.Q5. For $\neg p \vee q$, what is the value on the row $p =$T, $q =$F?

19.1.Q6. Building $(p \wedge q) \vee r$, which sub-column should you compute first?

A. r

B. $p \wedge q$

C. $(p \wedge q) \vee r$

D. $\neg p$

19.1.Q7. For $p \wedge (q \vee r)$, is the row $p =$F, $q =$T, $r =$T true or false?

19.1.Q8. True or False: every row of a truth table represents one possible assignment of T/F to the simple statements.

A. True

B. False

🐍 From Concept to Code (Python)

You can make Python **print a truth table** by looping the inputs over `[True, False]` with nested `for` loops – one loop per variable. Each pass gives one row, and you just evaluate the logical expression with `and`/`or`/`not`. This is exactly the 2^n rows, generated automatically.

```
# Truth table for  p and (q or r)
print("p     q     r     result")
for p in [True, False]:
    for q in [True, False]:
        for r in [True, False]:
            result = p and (q or r)
            print(p, q, r, result)
```

```
output
p     q     r     result
True True True True
True True False True
True False True True
True False False False
False True True False
False True False False
False False True False
False False False False
```

Try it:

1. Adapt the loop to print the truth table for `(not p) or q`. (Two variables, so two nested loops, 4 rows.)
2. Add a counter that increments each row and print it, to confirm the loop produces exactly $2^3 = 8$ rows.

Answer Key – Chapter 19 – Truth Tables

19.1.Q1. 16 – that is 2^4.

19.1.Q2. T – inclusive "or" is true when both parts are true.

19.1.Q3. Only the row p =T, q =T – a conjunction needs both parts true.

19.1.Q4. False – all-T is a **tautology**; all-F is a contradiction.

19.1.Q5. F – $\neg p$ is F there and q is F, so F∨F = F.

19.1.Q6. $p \wedge q$ – build the innermost bracket first, then combine with r.

19.1.Q7. False – since p is F, the whole conjunction is F regardless of $q \vee r$.

19.1.Q8. True – that is exactly what a row is, and listing all of them is why there are 2^n.

Chapter 20 · Conditionals & Implication

How to read and build the conditional $p \Rightarrow q$, work out its truth table, and form its converse, inverse, contrapositive and biconditional.

The Idea Behind It

"If p then q" is false in only one case: p true while q false. *but why?*

Read $p \Rightarrow q$ as a **promise**: *"whenever p happens, q happens too."* The only way to be caught breaking that promise is for p to come true while q fails to, on every other combination you have kept your word (and a promise about a situation that never arises counts as kept). That single broken case is exactly the meaning of *"not p, or else q"*, so the governing fact is the equivalence

$$\boxed{p \Rightarrow q \;\equiv\; \neg p \vee q}$$

To prove it, build a column for each side over all $2^2 = 4$ rows and compare:

p	q	$\neg p$	$\neg p \vee q$	$p \Rightarrow q$
T	T	F	T	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

The two right-hand columns read `T, F, T, T` identically, so the equivalence holds, and both are F on exactly the single row $p = \mathsf{T},\ q = \mathsf{F}$.

$$\therefore\ p \Rightarrow q \text{ is false } \text{only when } p \text{ is true and } q \text{ is false}$$

"If it rains I bring an umbrella" is broken only on the day it rains and you turn up without one, not by a dry day. And the bottom two rows show a false hypothesis makes the conditional automatically (**vacuously**) true.

only a true premise can break it

Almost every rule, promise and law has the shape *"if … then …"*. "If you score 40%, then you pass." "If it is a square, then it has four equal sides." This *if–then* structure is the **conditional**, and it is the connective that powers mathematical proof and every line of code that makes a decision. It behaves in one way that surprises newcomers, so we will be careful.

CONDITIONAL / IMPLICATION ($\IMP$)

The **conditional** $p \Rightarrow q$, read *"if p then q"* or *"p implies q"*, is **false in exactly one case: when p is true but q is false**. In every other case it is true. Here p is the **hypothesis** (or antecedent) and q is the **conclusion** (or consequent).

The key is to read $p \Rightarrow q$ as a **promise**: "whenever p happens, q will happen too." The only way to **break** that promise is for p to come true while q fails. So $p \Rightarrow q$ is false precisely on the row $p =$T, $q =$F.

The truth table of the conditional

p	q	$p \Rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

Look hard at the bottom two rows, where p is **false**. The conditional is **true** there, no matter what q is. This is called a **vacuously true** statement: the promise "if p then q" was never put to the test, because p did not happen, so it counts as kept. "If you score 40%, you pass" is *not broken* by a student who scored 30% – the rule said nothing about that student. A conditional with a false hypothesis is automatically true.

Note: $p \Rightarrow q$ does **not** claim that p causes q, nor that p is true. It only rules out the one combination "p true, q false." This is why a true conditional can have a false hypothesis – and why everyday cause-and-effect intuition can mislead you here.

Converse, inverse and contrapositive

From one conditional $p \Rightarrow q$ we can manufacture three close relatives by swapping and/or negating the two parts. Keeping them straight is essential.

> **THE THREE RELATIVES**
>
> Starting from the **conditional** $p \Rightarrow q$: - **Converse:** $q \Rightarrow p$ (swap the two parts) - **Inverse:** $\neg p \Rightarrow \neg q$ (negate both parts) - **Contrapositive:** $\neg q \Rightarrow \neg p$ (swap **and** negate)

The single most important fact about these: a conditional and its **contrapositive always have the same truth value** – they are logically equivalent. But a conditional and its **converse are *not* equivalent**, and neither are a conditional and its inverse. Let us see this in a table.

p	q	$p \Rightarrow q$	converse $q \Rightarrow p$	inverse $\neg p \Rightarrow \neg q$	contrapositive $\neg q \Rightarrow \neg p$
T	T	T	T	T	T
T	F	F	T	T	F
F	T	T	F	F	T
F	F	T	T	T	T

Compare columns. The **conditional** and the **contrapositive** match in every row – equivalent. The **converse** and the **inverse** also match each other (they are contrapositives of one another), but they do **not** match the original

conditional. So "if it rains, the ground is wet" does **not** let you conclude "if the ground is wet, it rained" (the converse – the ground could be wet from a sprinkler).

The biconditional – "if and only if"

BICONDITIONAL ($\EQV$)

The **biconditional** $p \Leftrightarrow q$, read *"p if and only if q"* (often shortened to *"iff"*), is **true exactly when p and q have the same truth value** – both true or both false. It asserts the conditional **both ways**: $p \Rightarrow q$ **and** $q \Rightarrow p$.

p	q	$p \Leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

A biconditional is the language of **definitions** and exact equivalences: "a number is even $\Leftrightarrow$ it is divisible by 2." Both directions hold, so the two statements stand or fall together.

Worked examples

Example 1

For the conditional "If $n = 4$, then n is even" applied to $n = 3$, is the statement true or false?

SOLUTION

1. Identify the parts for $n = 3$: hypothesis p is "$3 = 4$" (F); conclusion q is "3 is even" (F). *plug in n=3*
2. The conditional is false **only** on the row $p = T$, $q = F$. Here $p = F$. *wrong row*
3. A false hypothesis makes it vacuously true – the rule made no claim about $n = 3$. *promise untested*

$p = F \Rightarrow (p \Rightarrow q) = T$ **(vacuously true).**

Example 2

Write the converse, inverse and contrapositive of "If it is a square, then it is a rectangle." Which are true?

SOLUTION

1. Let p: "it is a square," q: "it is a rectangle." The original $p \Rightarrow q$ is **true**.
2. Converse $q \Rightarrow p$: "If it is a rectangle, then it is a square." False – a 2×3 rectangle is not a square.

   swap only
3. Inverse $\neg p \Rightarrow \neg q$: "If it is not a square, then it is not a rectangle." False – same rectangle is a counterexample. *negate only*
4. Contrapositive $\neg q \Rightarrow \neg p$: "If it is not a rectangle, then it is not a square." True – equivalent to the original. *swap & negate*

Converse $q \Rightarrow p$: false. Inverse $\neg p \Rightarrow \neg q$: false. Contrapositive $\neg q \Rightarrow \neg p$: true.

Example 3

Given that "If the alarm rings, then there is a fire" is true, and the alarm rings, what can you conclude? What if instead there is no fire?

SOLUTION

1. The conditional $p \Rightarrow q$ is true and p (alarm rings) is true.
2. A true $p \Rightarrow q$ with p forces q: **there is a fire**. *modus ponens*
3. If instead $\neg q$ (no fire), the contrapositive $\neg q \Rightarrow \neg p$ gives $\neg p$: **the alarm did not ring**. *modus tollens*

Alarm rings $\Rightarrow$ fire. No fire $\Rightarrow$ alarm silent.

Example 4

Build the truth table for the biconditional $p \Leftrightarrow q$ by combining $p \Rightarrow q$ and $q \Rightarrow p$ with $\wedge$, and confirm it is true exactly when p and q agree.

SOLUTION

1. Form the two conditionals $p \Rightarrow q$ and $q \Rightarrow p$ in helper columns. *both directions*
2. Combine them: $(p \Rightarrow q) \wedge (q \Rightarrow p) = p \Leftrightarrow q$.
3. The result is T on rows TT and FF – exactly where p and q match. *same value*

Final column $p \Leftrightarrow q$ is T only when p and q agree.

p	q	$p \Rightarrow q$	$q \Rightarrow p$	$(p \Rightarrow q) \wedge (q \Rightarrow p)$
T	T	T	T	T
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T

Example 5

Restate "Only if you have a ticket may you enter" as a clean conditional. Be careful about direction.

SOLUTION

1. " A only if B " means A can happen only when B holds – so $A \Rightarrow B$. *only if → right*
2. Here A = "you enter," B = "you have a ticket." Thus entering implies having a ticket.
3. In symbols: $(\text{enter}) \Rightarrow (\text{ticket})$. It does **not** say a ticket guarantees entry – that is the converse.

not the other way

$(\text{you enter}) \Rightarrow (\text{you have a ticket})$.

Example 6

Build the truth table for $(p \wedge q) \Rightarrow p$ and decide whether it is a tautology.

SOLUTION

1. $2^2 = 4$ rows. Build the hypothesis $p \wedge q$ in a helper column first (true only on row TT). *hypothesis first*
2. Form the conditional $(p \wedge q) \Rightarrow p$: it is false **only** where the hypothesis is T but p is F – and that never happens, since $p \wedge q$ being T forces p to be T. *T hyp needs p true*
3. Every row of the final column is T, so the statement is a tautology. *all T → always true*

All four rows give T – a tautology ("p and q" certainly implies "p").

p	q	$p \wedge q$	$(p \wedge q) \Rightarrow p$
T	T	T	T
T	F	F	T
F	T	F	T
F	F	F	T

Example 7

Write the negation of "If it rains, then the match is cancelled," then confirm with a truth table that $\neg(p \Rightarrow q) \equiv p \wedge \neg q$.

SOLUTION

1. A conditional breaks only when the hypothesis holds but the conclusion fails. So $\neg(p \Rightarrow q)$ is "it rains and the match is not cancelled" $= p \wedge \neg q$. *breaks on T, F*
2. Verify by table: build $p \Rightarrow q$, then $\neg(p \Rightarrow q)$, and separately $p \wedge \neg q$. *compare two columns*
3. Both result columns read F, T, F, F – identical, so the negation really is $p \wedge \neg q$. *columns match*

"It rains and the match is not cancelled," i.e. $p \wedge \neg q$.

p	q	$p \Rightarrow q$	$\neg(p \Rightarrow q)$	$p \wedge \neg q$
T	T	T	F	F
T	F	F	T	T
F	T	T	F	F
F	F	T	F	F

Example 8

A rule states: "If a number is divisible by 6, then it is divisible by 3." Use the contrapositive to test $n = 20$.

SOLUTION

1. Let p: "n is divisible by 6," q: "n is divisible by 3." The rule is the true conditional $p \Rightarrow q$.
2. For $n = 20$: is it divisible by 3? No, so q is F, giving $\neg q$ is T. *20 not ÷ by 3*
3. The contrapositive $\neg q \Rightarrow \neg p$ is equivalent to the rule, and it forces $\neg p$: 20 is **not** divisible by 6. *contrapositive*
4. Check directly: $20 \div 6$ is not a whole number, confirming $\neg p$. *sanity check*

Since 20 is not divisible by 3, the contrapositive gives 20 is not divisible by 6.

Tip: To negate a conditional, remember it fails on a single row. The negation of $p \Rightarrow q$ is $p \wedge \neg q$ – "p happened **and** q did not." The opposite of a promise is not another conditional; it is the one situation that breaks it.

conditional hypothesis vacuously true converse contrapositive biconditional

⚠ Common Mistakes

- Confusing the **converse** with the **contrapositive**. The contrapositive ($\neg q \Rightarrow \neg p$) is equivalent to the original; the converse ($q \Rightarrow p$) is **not** – assuming it is the classic logical error.
- Thinking a conditional with a **false hypothesis** is false or "undefined." It is **true** (vacuously) – false only on the single row $p =$T, $q =$F.
- Reading $p \Rightarrow q$ as cause-and-effect or as claiming p is true. It only forbids "p true, q false"; it makes no causal or existence claim.
- Negating a conditional into another conditional. The negation of $p \Rightarrow q$ is $p \wedge \neg q$, not $p \Rightarrow \neg q$.
- Mis-translating "A only if B." It means $A \Rightarrow B$ (A forces B), not $B \Rightarrow A$.

📝 Quiz

20.1.Q1. On which single row is $p \Rightarrow q$ false?

A. p=T, q=T

B. p=T, q=F

C. p=F, q=T

D. p=F, q=F

20.1.Q2. True or False: "If $2 = 5$, then the Sun is hot" is a true statement.

A. True

B. False

20.1.Q3. Which is always logically equivalent to $p \Rightarrow q$?

A. Its converse $q \Rightarrow p$

B. Its inverse $\neg p \Rightarrow \neg q$

C. Its contrapositive $\neg q \Rightarrow \neg p$

D. None of these

20.1.Q4. Give the converse of "If it is raining, then the ground is wet."

20.1.Q5. When is the biconditional $p \Leftrightarrow q$ true?

20.1.Q6. The conditional "If you study, you pass" is true, and you did **not** pass. By the contrapositive, what follows?

20.1.Q7. True or False: the negation of $p \Rightarrow q$ is $p \Rightarrow \neg q$.

A. True

B. False

20.1.Q8. Rewrite "You may board only if you have a boarding pass" as a conditional.

🐍 From Concept to Code (Python)

Python's **`if / elif / else`** is the conditional in action: `if p:` runs a block exactly when the statement `p` is true. You can also evaluate $p \Rightarrow q$ directly – since "false only when p is true and q is false," it equals `(not p) or q`.

```
def implies(p, q):
    return (not p) or q       # p -> q, the conditional

print(implies(True,  True))   # T -> T
print(implies(True,  False))  # the only False case
print(implies(False, False))  # vacuously true

score = 30
if score >= 40:
    print("pass")
else:
    print("fail")
```

```
output
True
False
True
fail
```

Try it:

1. Use the `implies` function to print the full truth table of $p \Rightarrow q$ with two nested loops over `[True, False]`.
2. Write `contrapositive(p, q)` returning `implies(not q, not p)` and check it gives the same column as `implies(p, q)` on all four rows.

Answer Key – Chapter 20 – Conditionals & Implication

20.1.Q1. p=T, q=F – a true hypothesis with a false conclusion is the only way to break the promise.

20.1.Q2. True – the hypothesis "$2 = 5$" is false, so the conditional is vacuously true.

20.1.Q3. The contrapositive $\neg q \Rightarrow \neg p$ – it matches the original in every row.

20.1.Q4. "If the ground is wet, then it is raining." (Not equivalent to the original – the ground could be wet from other causes.)

20.1.Q5. Exactly when p and q have the **same** truth value (both T or both F).

20.1.Q6. You did not study – by *modus tollens* / the contrapositive $\neg(\text{pass}) \Rightarrow \neg(\text{study})$.

20.1.Q7. False – the negation is $p \wedge \neg q$ (the single case that breaks the conditional).

20.1.Q8. (you board) $\Rightarrow$ (you have a boarding pass) – "only if" puts the condition on the right.

Chapter 21 · Equivalence & De Morgan's Laws

How to show two statements are logically equivalent ($\equiv$) with a truth table, and how to apply De Morgan's laws to negate AND/OR.

The Idea Behind It

The opposite of "both" is "at least one is missing": $\neg(p \wedge q) \equiv \neg p \vee \neg q$. *but why?*

To deny "p **and** q" is to deny that the pair holds *together*. The only way they fail to hold together is for at least one of them to be false, that is, "p is false **or** q is false." So negating an AND should flip it to an OR of the negations, **De Morgan's first law**:

$$\boxed{\neg(p \wedge q) \equiv \neg p \vee \neg q}$$

Equivalence is a precise claim, the two columns must match on **every** row, so prove it mechanically over all $2^2 = 4$ rows:

p	q	$p \wedge q$	$\neg(p \wedge q)$	$\neg p$	$\neg q$	$\neg p \vee \neg q$
T	T	T	F	F	F	F
T	F	F	T	F	T	T
F	T	F	T	T	F	T
F	F	F	T	T	T	T

The two highlighted columns, $\neg(p \wedge q)$ and $\neg p \vee \neg q$, both read F, T, T, T, agreeing in every row.

$$\therefore \neg(p \wedge q) \equiv \neg p \vee \neg q$$

Notice the punchline: negation **flips the connective**, AND becomes OR. The dual law, found the same way, is $\neg(p \vee q) \equiv \neg p \wedge \neg q$ ("the opposite of *at least one* is *neither*"). *not both = at least one missing*

Two statements can look completely different yet behave **identically** in every possible situation. "It is not the case that the door is closed" and "the door is open" are written differently but mean the same thing. When two compound statements share the same truth value on *every* row of the truth table, we call them **logically equivalent** – and we may freely swap one for the other. This is how we *simplify* logic, the exact counterpart of simplifying an algebraic expression.

LOGICAL EQUIVALENCE ($\EQUIV$)

Two statements are **logically equivalent**, written $A \equiv B$, when their truth-table columns are **identical** – they have the same truth value for every assignment of their simple statements. Equivalent statements are interchangeable.

Take care to keep two symbols apart. The **biconditional** $p \Leftrightarrow q$ is itself a *statement* (true on some rows, false on others). **Equivalence** $A \equiv B$ is a *claim about two statements* – that their columns match everywhere. The link:

$A \equiv B$ is true exactly when the biconditional $A \Leftrightarrow B$ is a **tautology** (true on every row).

Proving an equivalence by truth table

The method is mechanical and certain: build one truth table containing a column for A and a column for B. If the two columns agree in **every** row, then $A \equiv B$. If they differ in even one row, they are **not** equivalent. Let us warm up with the simplest equivalence of all.

Double negation

Double negation: $\neg(\neg p) \equiv p$. Negating twice returns the original – the table makes it obvious:

p	$\neg p$	$\neg(\neg p)$
T	F	T
F	T	F

The $\neg(\neg p)$ column matches the p column exactly, so $\neg(\neg p) \equiv p$.

De Morgan's Laws

The two **De Morgan's laws** are the most useful equivalences in all of logic. They tell you how a negation **distributes** across *and* and *or* – and the punchline is that it **flips the connective**:

$$\neg(p \wedge q) \equiv \neg p \vee \neg q$$
$$\neg(p \vee q) \equiv \neg p \wedge \neg q$$

> **DE MORGAN'S LAWS**
>
> To negate a conjunction or disjunction, **negate each part and swap the connective**: - "**not** (p **and** q)" $\equiv$ "(**not** p) **or** (**not** q)" - "**not** (p **or** q)" $\equiv$ "(**not** p) **and** (**not** q)"

In words: the opposite of "both" is "at least one is *not*", and the opposite of "at least one" is "*neither*" (both are *not*). "It is **not** the case that it is hot **and** sunny" means "it is **not** hot **or** it is **not** sunny." We will prove the first law in full below.

Proof of the first De Morgan law

Build columns for the left side $\neg(p \wedge q)$ and the right side $\neg p \vee \neg q$, then compare:

p	q	$p \wedge q$	$\neg(p \wedge q)$	$\neg p$	$\neg q$	$\neg p \vee \neg q$
T	T	T	F	F	F	F
T	F	F	T	F	T	T
F	T	F	T	T	F	T
F	F	F	T	T	T	T

The two highlighted result columns – $\neg(p \wedge q)$ and $\neg p \vee \neg q$ – read **F, T, T, T** identically. Equal in every row, so $\neg(p \wedge q) \equiv \neg p \vee \neg q$. ∎

A small toolbox of equivalences

Beyond double negation and De Morgan, a few more equivalences are worth knowing; each can be verified by the same truth-table method.

- **Commutative:** $p \wedge q \equiv q \wedge p$ and $p \vee q \equiv q \vee p$ (order doesn't matter).
- **Distributive:** $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$, and $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$.
- **Conditional as disjunction:** $p \Rightarrow q \equiv \neg p \vee q$ (a conditional rewritten without "⇒").
- **Contrapositive:** $p \Rightarrow q \equiv \neg q \Rightarrow \neg p$ (from the previous chapter).

Worked examples

Example 1

Use a truth table to decide whether $\neg(p \vee q)$ is equivalent to $\neg p \wedge \neg q$ (the second De Morgan law).

SOLUTION

1. Build $p \vee q$, then negate it for the left side $\neg(p \vee q)$.
2. Separately build $\neg p$ and $\neg q$, then $\neg p \wedge \neg q$ for the right side. *negate each, AND*
3. Compare the two result columns: both read F, F, F, T – identical. *every row agrees*

Yes – they match in every row, so $\neg(p \vee q) \equiv \neg p \wedge \neg q$.

p	q	$p \vee q$	$\neg(p \vee q)$	$\neg p \wedge \neg q$
T	T	T	F	F
T	F	T	F	F
F	T	T	F	F
F	F	F	T	T

Example 2

Negate "It is hot **and** it is sunny" and rewrite it in plain English using De Morgan.

SOLUTION

1. Let p: "it is hot," q: "it is sunny." We want $\neg(p \wedge q)$.
2. By the first De Morgan law, $\neg(p \wedge q) \equiv \neg p \vee \neg q$. *AND flips to OR*
3. Translate back: not hot, or not sunny. (At least one of the two fails.)

"It is not hot, or not sunny."

Example 3

Negate "Every light is off **or** the alarm is set." (Let p: every light is off, q: the alarm is set.)

SOLUTION

1. We want $\neg(p \vee q)$.
2. Second De Morgan law: $\neg(p \vee q) \equiv \neg p \wedge \neg q$. OR flips to AND
3. In words: "not every light is off **and** the alarm is not set," i.e. a light is on and the alarm is not set.

"**At least one light is on, and the alarm is not set.**"

Example 4

Simplify $\neg(\neg p \wedge q)$ as far as you can using De Morgan and double negation.

SOLUTION

1. Apply De Morgan to the conjunction inside: $\neg(\neg p \wedge q) \equiv \neg(\neg p) \vee \neg q$. AND → OR
2. Use double negation on the first term: $\neg(\neg p) \equiv p$. two bars cancel
3. So the whole thing collapses to $p \vee \neg q$.

$\neg(\neg p \wedge q) \equiv p \vee \neg q$.

Example 5

Show by truth table that $p \Rightarrow q$ is equivalent to $\neg p \vee q$.

SOLUTION

1. Build the conditional $p \Rightarrow q$ (false only on the row p=T, q=F).
2. Build $\neg p \vee q$ separately. drops the arrow
3. Both columns read T, F, T, T – identical, so $p \Rightarrow q \equiv \neg p \vee q$. Now any "⇒" can be rewritten with ¬, ∨ .

Equivalent – both give T, F, T, T, so $p \Rightarrow q \equiv \neg p \vee q$.

p	q	$p \Rightarrow q$	$\neg p$	$\neg p \vee q$
T	T	T	F	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

Example 6

Verify the distributive law $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$ by truth table.

SOLUTION

1. $2^3 = 8$ rows. For the left side, build $q \vee r$ then $p \wedge (q \vee r)$. *inside-out, left side*
2. For the right side, build $p \wedge q$ and $p \wedge r$, then $(p \wedge q) \vee (p \wedge r)$. *each AND, then OR*
3. Compare the two result columns: both read T, T, T, F, F, F, F, F – identical in every row, so the equivalence holds. *all 8 rows agree*

Equivalent – the two columns match on all 8 rows, so $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$.

p	q	r	$q \vee r$	$p \wedge (q \vee r)$	$(p \wedge q) \vee (p \wedge r)$
T	T	T	T	T	T
T	T	F	T	T	T
T	F	T	T	T	T
T	F	F	F	F	F
F	T	T	T	F	F
F	T	F	T	F	F
F	F	T	T	F	F
F	F	F	F	F	F

Example 7

Simplify $\neg(p \vee \neg q)$ as far as possible using De Morgan and double negation.

SOLUTION

1. Apply De Morgan to the disjunction inside: negate each part and flip OR to AND: $\neg(p \vee \neg q) \equiv \neg p \wedge \neg(\neg q)$. *OR → AND*
2. The second term has two negations on q; double negation cancels them: $\neg(\neg q) \equiv q$. *two bars cancel*
3. So the whole expression collapses to $\neg p \wedge q$. *push ¬ inward, then simplify*

$\neg(p \vee \neg q) \equiv \neg p \wedge q$.

Example 8

Negate the conditional-containing statement $p \wedge (q \Rightarrow r)$, simplifying so every $\neg$ sits on a simple statement.

SOLUTION

1. First De Morgan on the outer $\wedge$ (flip to OR, negate each part): $\neg(p \wedge (q \Rightarrow r)) \equiv \neg p \vee \neg(q \Rightarrow r)$.
   AND → OR
2. Rewrite the negated conditional. The negation of a conditional is $\neg(q \Rightarrow r) \equiv q \wedge \neg r$ (the one case that breaks it). ¬ of ⇒ = $q \wedge \neg r$
3. Substitute back: $\neg p \vee (q \wedge \neg r)$. Every $\neg$ now sits on a single letter. fully pushed in

$\neg(p \wedge (q \Rightarrow r)) \equiv \neg p \vee (q \wedge \neg r)$.

Note: De Morgan's laws are everywhere in programming. The condition "stop unless the user is logged in **and** verified" negates to "stop if **not** logged in **or not** verified." Getting the connective flip right (AND becomes OR) is a frequent source of bugs – and the reason this chapter matters well beyond the page.

Tip: To negate any expression quickly, push the negation **inward**: each $\neg$ that crosses a bracket flips $\wedge \leftrightarrow \vee$ and negates what is inside, until every $\neg$ sits on a simple statement. Two negations on the same statement cancel. This is just repeated De Morgan plus double negation.

logically equivalent De Morgan's laws double negation distributive law

⚠ Common Mistakes

- Forgetting to **swap the connective** in De Morgan. $\neg(p \wedge q)$ becomes $\neg p \vee \neg q$ (AND turns into OR), **not** $\neg p \wedge \neg q$.
- Negating only the parts but leaving the connective alone, e.g. writing $\neg(p \vee q)$ as $\neg p \vee \neg q$. The correct form is $\neg p \wedge \neg q$.
- Confusing **equivalence** $\equiv$ (a claim that two columns match everywhere) with the **biconditional** $\Leftrightarrow$ (itself a statement, true on some rows, false on others).
- Stopping a simplification too early – after a De Morgan step, look for a **double negation** like $\neg(\neg p)$ that can still collapse to p.
- Declaring two statements equivalent after checking only some rows. Equivalence requires agreement on **every** row of the full 2^n-row table.

Quiz

21.1.Q1. De Morgan: $\neg(p \wedge q)$ is equivalent to which of these?

A. $\neg p \wedge \neg q$

B. $\neg p \vee \neg q$

C. $p \vee q$

D. $\neg p \Rightarrow q$

21.1.Q2. Rewrite $\neg(p \vee q)$ without the outer negation.

21.1.Q3. True or False: $A \equiv B$ means A and B have the same truth value in every row of the truth table.

A. True

B. False

21.1.Q4. Simplify $\neg(\neg p)$.

21.1.Q5. Negate "It is raining and cold" in plain English.

21.1.Q6. Which statement is logically equivalent to the conditional $p \Rightarrow q$?

A. $p \wedge \neg q$

B. $\neg p \vee q$

C. $q \Rightarrow p$

D. $\neg p \wedge q$

21.1.Q7. True or False: $\neg(p \vee q)$ is equivalent to $\neg p \vee \neg q$.

A. True

B. False

21.1.Q8. Simplify $\neg(p \wedge \neg q)$ using De Morgan and double negation.

21.1.Q9. How many rows of the truth table must agree for two statements in p, q, r to be equivalent?

A. 3

B. 6

C. All 8

D. Any one

🐍 From Concept to Code (Python)

Equivalences let you **simplify conditions** in code so they read clearly. De Morgan in Python: `not (a and b)` equals `(not a) or (not b)`. You can *prove* an equivalence by checking it on every combination with a loop – if it is `True` on all rows, the two expressions are equivalent. Python's `and`/`or` also **short-circuit** (stop as soon as the answer is known), which is why ordering conditions can matter.

```python
# Verify De Morgan:  not (p and q) == (not p) or (not q)
all_match = True
for p in [True, False]:
    for q in [True, False]:
        left  = not (p and q)
        right = (not p) or (not q)
        if left != right:
            all_match = False
print("De Morgan holds:", all_match)
```

```
output
De Morgan holds: True
```

Try it:

1. Use the same loop pattern to verify double negation: `not (not p) == p` for `p` in `[True, False]`.
2. Rewrite the condition `if not (age >= 18 and has_id):` using De Morgan so the `not` is pushed onto each part, and confirm it behaves the same.

Answer Key – Chapter 21 – Equivalence & De Morgan's Laws

21.1.Q1. $\neg p \vee \neg q$ – negate each part and switch AND to OR.

21.1.Q2. $\neg p \wedge \neg q$ – negate each part and switch OR to AND.

21.1.Q3. True – that is the definition of logical equivalence.

21.1.Q4. p – double negation cancels the two $\neg$ signs.

21.1.Q5. "It is not raining or it is not cold" (De Morgan: AND becomes OR).

21.1.Q6. $\neg p \vee q$ – proven equal by truth table (both give T, F, T, T).

21.1.Q7. False – that keeps OR; the correct De Morgan form swaps to AND: $\neg p \wedge \neg q$.

21.1.Q8. $\neg p \vee q$ – De Morgan gives $\neg p \vee \neg(\neg q)$, and $\neg(\neg q) = q$.

21.1.Q9. All 8 $(= 2^3)$ rows must agree – equivalence is checked on every assignment.

Chapter 22 · Quantifiers: For All & There Exists

*How to read, write and **negate** statements that say "for every..." or "there is at least one...", and how a single counterexample can topple a sweeping claim.*

The Idea Behind It

To deny "for all x, $P(x)$" you need just one counterexample: $\neg(\forall x\, P(x)) \equiv \exists x\, \neg P(x)$. *but why?*

The claim $\forall x\, P(x)$ says P holds *everywhere*, with **no exceptions**. So the claim is wrong exactly when an exception exists, a single x where P fails. "There is no exception is false" means "there is an exception", which is precisely $\exists x\, \neg P(x)$. That gives the negation rule:

$$\boxed{\neg\big(\forall x\, P(x)\big) \;\equiv\; \exists x\, \neg P(x)}$$

Check both directions, since the domain can be infinite a truth table will not fit, so argue by cases:

- If $\forall x\, P(x)$ is **false**, then "P holds for every x" fails, so some x must break it: that x satisfies $\neg P(x)$, giving $\exists x\, \neg P(x)$.
- Conversely, if some x has $\neg P(x)$, that witness is a genuine exception, so $\forall x\, P(x)$ cannot stand.

Each side forces the other, so they are equivalent.

$$\therefore\ \text{one counterexample}\ \text{disproves a}\ \forall\ \text{claim}$$

"All swans are white" is toppled by spotting **one** non-white swan, you need not survey the rest. The dual works the same way: $\neg(\exists x\, P(x)) \equiv \forall x\, \neg P(x)$ ("no x works" means "every x fails"). Negation just **swaps the quantifier** and negates inside. *find the single exception*

So far every statement has been about one fixed thing: "7 is prime", "it is raining". But most of mathematics talks about *whole collections* at once – "**every** even number is divisible by 2", "**some** number squared equals 2". The little words **every**, **all**, **some** and **there exists** are called **quantifiers**, and giving them precise symbols lets us state and disprove general claims without any guesswork.

UNIVERSAL QUANTIFIER

$\forall$ is read "**for all**" or "for every". The statement $\forall x,\ P(x)$ claims that $P(x)$ is true for *every* value of x in the collection we are discussing (the **domain**). One single value that fails makes the whole statement false.

EXISTENTIAL QUANTIFIER

$\exists$ is read "**there exists**" or "there is at least one". The statement $\exists x,\ P(x)$ claims that $P(x)$ is true for *at least one* value of x. One single value that works makes the whole statement true.

Here $P(x)$ is a **predicate** – a sentence with a variable in it, such as "$x > 0$", which only becomes true or false once you choose an x. Read the comma as "such that" or simply pause: $\forall n,\ n + 0 = n$ is "for all n, n plus zero equals n ".

The two quantifiers are opposite extremes. $\forall$ demands the predicate hold **everywhere**; $\exists$ asks only that it hold **somewhere**. Always keep the **domain** in mind – "$\exists x,\ x^2 = 2$" is *false* over the integers $\mathbb{Z}$ but *true* over the real numbers $\mathbb{R}$.

Counterexamples – disproving a "for all"

Because $\forall x,\ P(x)$ insists P holds for *every* x, you can knock it down by exhibiting **one** value where P fails. That value is a **counterexample**. Finding a single counterexample is a complete disproof – you do not need to say anything about the other cases.

universal quantifier existential quantifier predicate domain counterexample

> **Note:** Symmetry of effort: to **prove** $\forall x, P(x)$ you must argue about every x; to **disprove** it you need just one counterexample. To **prove** $\exists x, P(x)$ you need just one example (a "witness"); to **disprove** it you must rule out every x.

Negating a quantified statement

Negation **swaps the quantifier and negates the predicate**. This is the quantifier version of De Morgan's laws, and it is the single most useful rule in the chapter:

$$\neg(\forall x,\ P(x)) \ \equiv\ \exists x,\ \neg P(x)$$
$$\neg(\exists x,\ P(x)) \ \equiv\ \forall x,\ \neg P(x)$$

In words: the opposite of "**all** are" is "**some** is **not**", and the opposite of "**some** is" is "**none** is" (i.e. "all are **not**"). Saying "**not** everyone passed" is the same as "**someone** failed" – it does *not* mean "**everyone** failed", a trap we will meet again in the Mistakes box.

Worked examples

Example 1

Write "every natural number is greater than or equal to 0" in symbols, with domain $\mathbb{N}$.

SOLUTION

1. "every" signals the universal quantifier $\forall$. every → for all
2. The predicate is $n \geq 0$.
3. Combine them: $\forall n \in \mathbb{N},\ n \geq 0$.

$\forall n \in \mathbb{N},\ n \geq 0$ **– a true statement.**

Example 2

Write "some integer is negative" in symbols, with domain $\mathbb{Z}$, and say whether it is true.

SOLUTION

1. "Some" signals the existential quantifier $\exists$. *some → exists*
2. The predicate is $n < 0$.
3. In symbols: $\exists n \in \mathbb{Z},\ n < 0$.
4. It only needs one witness: $n = -5$ satisfies $-5 < 0$. *one is enough*

$\exists n \in \mathbb{Z},\ n < 0$ **– true, witnessed by** $n = -5$**.**

Example 3

Disprove the claim "$\forall n \in \mathbb{N},\ n^2 > n$".

SOLUTION

1. A "for all" claim falls to a single counterexample, so test small values. *hunt one failure*
2. Try $n = 0$: $0^2 = 0$, and $0 > 0$ is **false**.
3. Try $n = 1$: $1^2 = 1$, and $1 > 1$ is also **false**.
4. Either value breaks the claim; one is enough.

False. Counterexample: $n = 1$ **gives** $1^2 = 1 \not> 1$**.**

Example 4

Negate "$\forall x \in \mathbb{R},\ x^2 \geq 1$" and state which (original or negation) is true.

SOLUTION

1. Apply the rule: swap $\forall$ to $\exists$ and negate the predicate. *flip & negate*
2. Negating $x^2 \geq 1$ gives $x^2 < 1$.
3. Negation: $\exists x \in \mathbb{R},\ x^2 < 1$.
4. The negation is true (e.g. $x = 0$ gives $0 < 1$), so the original is false.

Negation: $\exists x \in \mathbb{R},\ x^2 < 1$ **– and *it* is the true one.**

Example 5

Negate the English sentence "All swans are white."

SOLUTION

1. Identify the structure: $\forall$ swan s, $\text{white}(s)$.
2. Swap to $\exists$ and negate "is white" to is not white. *all → some not*
3. Translate back to natural English.

"**Some swan is not white**" – i.e. $\exists s,\ \neg\text{white}(s)$. **(Not "no swan is white".)**

Example 6

A claim uses two quantifiers: "$\forall x \in \mathbb{R},\ \exists y \in \mathbb{R},\ y > x$" (every real has a bigger real). Negate it.

SOLUTION

1. Negate left to right, flipping each quantifier and finally the predicate. *flip every one*
2. $\forall x$ becomes $\exists x$; the inner $\exists y$ becomes $\forall y$.
3. The predicate $y > x$ negates to $y \leq x$.
4. Assemble in the same order.

$\exists x \in \mathbb{R},\ \forall y \in \mathbb{R},\ y \leq x$ – **"some real is $\geq$ every real". (False, as the original is true.)**

Example 7

Decide the truth of "$\exists n \in \mathbb{N},\ \forall m \in \mathbb{N},\ n \leq m$" (there is a natural number $\leq$ every natural number).

SOLUTION

1. Read the order strictly: one fixed n must beat *every* m. We hunt a single witness n. *one n for all m*
2. Try $n = 0$: is $0 \leq m$ for every $m \in \mathbb{N}$? Each natural number is ≥ 0, so $0 \leq m$ holds for all m.
3. The witness $n = 0$ satisfies the inner "for all", so the whole statement is true. *0 is the smallest*

True – **witnessed by $n = 0$, the least natural number.**

Example 8

Find a counterexample to " $\forall x \in \mathbb{R},\ x^2 > x$ ", paying attention to the *whole* real line.

SOLUTION

1. A "for all" over $\mathbb{R}$ falls to one bad x, so probe values *between* 0 and 1, where squaring shrinks a number.

   square can shrink

2. Try $x = \frac{1}{2}$: then $x^2 = \frac{1}{4}$, and $\frac{1}{4} > \frac{1}{2}$ is **false**.
3. One value breaks the claim; the rule "squaring enlarges" fails on $(0, 1)$.

False. Counterexample: $x = \frac{1}{2}$ **gives** $x^2 = \frac{1}{4} \not> \frac{1}{2}$**.**

Example 9

Negate the layered claim " $\forall \varepsilon > 0,\ \exists \delta > 0,\ \delta < \varepsilon$ ", then say which version is true.

SOLUTION

1. Work outward to inward, flipping each quantifier in turn and negating the predicate last.

   flip $\forall$, flip $\exists$, negate

2. The outer $\forall \varepsilon$ becomes $\exists \varepsilon$; the inner $\exists \delta$ becomes $\forall \delta$.
3. The predicate $\delta < \varepsilon$ negates to $\delta \geq \varepsilon$.
4. Reassemble in order: $\exists \varepsilon > 0,\ \forall \delta > 0,\ \delta \geq \varepsilon$.
5. The *original* is true (given any ε, pick $\delta = \varepsilon/2$), so the negation is false.

Negation: $\exists \varepsilon > 0,\ \forall \delta > 0,\ \delta \geq \varepsilon$ **– and the original (not this) is the true one.**

Note: Order matters when quantifiers mix. "$\forall x\, \exists y$" (everyone has *their own* matching y) is much weaker than " $\exists y\, \forall x$" (one single y works for everyone). Read them strictly left to right.

To negate any quantified sentence in your head, run a three-step flip: **(1)** change every $\forall$ to $\exists$ and every $\exists$ to $\forall$, **(2)** keep the variables and their order exactly as they are, **(3)** negate only the final predicate. "Every dog barks" $\rightarrow$ flip to "some dog" $\rightarrow$ negate to "does **not** bark" $\rightarrow$ "**some dog does not bark**". The mirror-words help too: the opposite of *all* is *some…not*, and the opposite of *some* is *none*.

⚠ Common Mistakes

- Negating "all are" as "none are". The true negation of "all students passed" is "**at least one** student did not pass", **not** "no student passed".
- Trying to prove a $\forall$ statement by checking a few examples. Examples can only *disprove* (one counterexample) a universal claim; proving it needs a general argument covering every case.
- Forgetting the domain. "$\exists x,\ x^2 = 2$" is false over $\mathbb{Z}$ but true over $\mathbb{R}$ – the same sentence changes truth value when the domain changes.
- Swapping the order of mixed quantifiers. "$\forall x\, \exists y,\ y > x$" (true for reals) is **not** the same as "$\exists y\, \forall x,\ y > x$" (false – no single number beats them all).
- Reading $\exists$ as "exactly one". $\exists$ means "**at least** one"; having two or more witnesses still makes it true.

Quiz

22.1.Q1. How is the symbol $\forall$ read?

A. there exists

B. for all

C. not

D. implies

22.1.Q2. What is the negation of "$\forall n,\ P(n)$"?

A. $\forall n,\ \neg P(n)$

B. $\exists n,\ \neg P(n)$

C. $\exists n,\ P(n)$

D. $\neg\exists n,\ P(n)$

22.1.Q3. Give a counterexample to "every prime number is odd".

22.1.Q4. True or False: a single counterexample is enough to disprove a "for all" statement.

A. True

B. False

22.1.Q5. True or False: "Not everyone passed the exam" means "everyone failed the exam".

A. True

B. False

22.1.Q6. Write "there is an even prime" in symbols over the domain of primes p.

22.1.Q7. Negate "$\exists x \in \mathbb{R},\ x^2 < 0$" and say which is true.

22.1.Q8. Which is the *weaker* statement: (A) $\exists y\, \forall x,\ y > x$ or (B) $\forall x\, \exists y,\ y > x$?

A. A

B. B

C. They are equal

D. Neither

🐍 From Concept to Code (Python)

Python turns quantifiers into one-line truth tests over a list. `all(...)` is ∀ – true only if **every** item passes. `any(...)` is ∃ – true if **at least one** passes. Each takes a sequence of `True`/`False` values, usually built with a *generator expression* `(condition for x in data)`.

```
nums = [2, 4, 6, 8]
print(all(n % 2 == 0 for n in nums))   # ∀ n: n is even?
print(any(n > 7 for n in nums))        # ∃ n: n > 7?
print(any(n < 0 for n in nums))        # ∃ n: n < 0?
```

```
output
True
True
False
```

```
# De Morgan for quantifiers:  not all  ==  any ... not
nums = [2, 4, 5, 8]
print(not all(n % 2 == 0 for n in nums))   # "not every n is even"
print(any(n % 2 != 0 for n in nums))       # "some n is odd"  (same!)
```

```
output
True
True
```

Try it:

1. Use `all()` to check whether every number in `[3, 6, 9, 12]` is divisible by 3.
2. A counterexample search: with `any()`, test whether the claim "$\forall n,\ n^2 > n$" fails for any `n` in `range(0, 5)`.

Answer Key – Chapter 22 – Quantifiers: For All & There Exists

22.1.Q1. "For all" (the universal quantifier).

22.1.Q2. $\exists n,\ \neg P(n)$ – swap the quantifier and negate the predicate.

22.1.Q3. 2 – it is prime but even, so it disproves the claim.

22.1.Q4. True – $\forall$ requires *every* case to hold, so one failure breaks it.

22.1.Q5. False – it means *at least one* person failed; others may still have passed.

22.1.Q6. $\exists p,\ p \text{ is even}$ – true, witnessed by $p = 2$.

22.1.Q7. $\forall x \in \mathbb{R},\ x^2 \geq 0$ – the negation is true; no real has a negative square.

22.1.Q8. B – "each x has its own bigger y" is far easier to satisfy than "one y beats every x".

Chapter 23 · Sets & Venn Diagrams

*How to describe a collection of objects as a **set**, test membership and subsets, combine sets with union, intersection and complement, and see why those three operations are exactly the logic words OR, AND and NOT.*

The Idea Behind It

Counting a union: $|A \cup B| = |A| + |B| - |A \cap B|$. Why the subtraction? *but why?*

When you add $|A| + |B|$ you are tallying every element of A and then every element of B. An element that lives in **both** sets (the overlap $A \cap B$) therefore gets counted **twice**, once in each tally, while an element in only one set is counted once. To fix the over-count you remove one copy of each shared element, that is, subtract $|A \cap B|$ a single time. Splitting the union into its three disjoint regions makes the bookkeeping exact:

Region	Counted by $\lvert A\rvert + \lvert B\rvert$	Should be
in A only	1 time	1
in B only	1 time	1
in both ($A \cap B$)	2 times	1

Only the overlap row is wrong, and it is over by exactly one each, counted twice but wanted once. Removing that surplus gives the count:

$$|A \cup B| = |A| + |B| - |A \cap B|$$

A quick check: $A = \{1, 2, 3, 4\}$, $B = \{3, 4, 5, 6\}$ share $\{3, 4\}$, so $4 + 4 - 2 = 6$, and indeed $A \cup B = \{1, 2, 3, 4, 5, 6\}$ has 6 elements. This "**don't double-count the overlap**" idea is the start of inclusion–exclusion. *subtract the shared part once*

A **set** is just a collection of distinct objects – the even numbers, the letters of your name, the students in a class. Sets are the bookkeeping system of mathematics: once you can say precisely *which things belong*, you can combine collections and reason about overlaps. And as we will see, the three core set operations are the very same AND/OR/NOT you met with truth tables, now acting on groups instead of single statements.

SET & ELEMENT

A **set** is an unordered collection of distinct **elements**. Write it inside braces, e.g. $A = \{1, 2, 3\}$. "$x \in A$" means "x is an element of A"; "$x \notin A$" means it is not. Order and repetition do not matter: $\{1, 2, 2, 3\}$ and $\{3, 1, 2\}$ are the same set as $\{1, 2, 3\}$.

Two special sets appear constantly. The **empty set** $\varnothing = \{\,\}$ has no elements at all. The **universal set** U is everything currently under discussion (for number work, perhaps $U = \mathbb{Z}$); every other set is drawn from inside it. Sets can be listed ($\{2, 4, 6\}$) or described by a rule, called **set-builder** notation: $\{\, x \mid x \text{ is even} \,\}$ reads "the set of all x such that x is even".

SUBSET

$A \subseteq B$ ("A is a **subset** of B") means *every* element of A is also in B. So $\{1, 2\} \subseteq \{1, 2, 3\}$. Note $A \subseteq A$ always, and $\varnothing \subseteq B$ for every B. If $A \subseteq B$ but $A \neq B$, we call A a **proper** subset, written $A \subsetneq B$.

Combining sets

Three operations build new sets from old ones. Read the defining rule on the right of each – notice the logic word hiding inside it:

UNION, INTERSECTION, COMPLEMENT

- **Union** $A \cup B = \{\, x \mid x \in A \textbf{ or } x \in B \,\}$ – everything in *either* set (the merge).
- **Intersection** $A \cap B = \{\, x \mid x \in A \textbf{ and } x \in B \,\}$ – only what is in *both* (the overlap).
- **Complement** $\complement A = A' = \{\, x \in U \mid x \notin A \,\}$ – everything in the universe that is **not** in A.

The link is exact and worth saying out loud: **union is OR, intersection is AND, complement is NOT.** Membership "$x \in A \cup B$" is the *statement* "$(x \in A) \vee (x \in B)$"; "$x \in A \cap B$" is "$(x \in A) \wedge (x \in B)$"; "$x \in \complement A$" is "$\neg(x \in A)$". Everything you proved with truth tables transfers straight to sets.

set union intersection complement

Venn diagrams – in words

A **Venn diagram** pictures sets as overlapping circles inside a rectangle (the universe U). Two circles A and B carve the rectangle into four regions; shading a region selects the elements it represents:

Region (two sets A, B)	Belongs to	Set expression
Inside A only (left petal)	A but not B	$A \cap \complement B$
The overlap (centre lens)	both A and B	$A \cap B$
Inside B only (right petal)	B but not A	$B \cap \complement A$
Outside both circles	neither	$\complement(A \cup B)$

So "$A \cup B$" shades **both circles entirely** (all three inner regions); "$A \cap B$" shades **only the centre lens**; "$\complement A$" shades **everything outside circle A**. Reading a diagram is just naming which of these regions are coloured.

Worked examples

Throughout, let $U = \{1, 2, 3, 4, 5, 6, 7, 8\}$, $A = \{1, 2, 3, 4\}$ and $B = \{3, 4, 5, 6\}$.

Example 1

List $A \cup B$.

SOLUTION

1. Union collects everything in either set. *union = OR*
2. From A: $1, 2, 3, 4$. From B: $3, 4, 5, 6$.
3. Merge and drop duplicates ($3, 4$ appear once).

$A \cup B = \{1, 2, 3, 4, 5, 6\}$

Example 2

List $A \cap B$.

SOLUTION

1. Intersection keeps only elements in both sets. *intersection = AND*
2. Check each element of A: is it also in B? 1 no, 2 no, 3 yes, 4 yes.

$A \cap B = \{3, 4\}$

Example 3

List $\complement A$ (complement within U).

SOLUTION

1. Complement keeps every element of the universe not in A. *complement = NOT*
2. $U = \{1, \ldots, 8\}$; remove $A = \{1, 2, 3, 4\}$. *cross out A*

$\complement A = \{5, 6, 7, 8\}$

Example 4

Is $\{3, 4\} \subseteq A$? Is $\{4, 5\} \subseteq A$?

SOLUTION

1. Subset requires every listed element to be in A. *check all*
2. $\{3, 4\}$: both $3 \in A$ and $4 \in A$ – yes.
3. $\{4, 5\}$: $4 \in A$ but $5 \notin A$ – one failure breaks it. *one miss = no*

$\{3, 4\} \subseteq A$ **is true;** $\{4, 5\} \subseteq A$ **is false.**

Example 5

Verify De Morgan for sets: show $\complement(A \cup B) = \complement A \cap \complement B$.

SOLUTION

1. Left side: $A \cup B = \{1, 2, 3, 4, 5, 6\}$, so $\complement(A \cup B) = \{7, 8\}$.
2. Right side: $\complement A = \{5, 6, 7, 8\}$ and $\complement B = \{1, 2, 7, 8\}$. union → intersection
3. Their intersection (elements in both) is $\{7, 8\}$.
4. Both sides land on the same set.

$\complement(A \cup B) = \complement A \cap \complement B = \{7, 8\}$ **– De Morgan confirmed.**

Example 6

In a class of 30, 18 study French (F) and 15 study Spanish (S); 7 study both. How many study neither?

SOLUTION

1. "Both" is the overlap: $|F \cap S| = 7$.
2. Avoid double-counting: $|F \cup S| = 18 + 15 - 7 = 26$. subtract overlap once
3. These 26 study at least one language; "neither" is the rest of the universe.
4. $30 - 26 = 4$.

4 **students study neither language.**

Example 7

Using set-builder notation, list $\{\, x \in U \mid x \text{ is even} \,\}$, then describe it as a single set operation with A and B.

SOLUTION

1. Read the rule: keep each x of the universe $U = \{1, \ldots, 8\}$ for which "x is even" is true.
   test the rule on each
2. Walk through U: $2, 4, 6, 8$ pass; $1, 3, 5, 7$ fail.
3. Compare with the given sets: $A = \{1, 2, 3, 4\}$ and $B = \{3, 4, 5, 6\}$ together hold $2, 4, 6$ but **miss** 8, so this even set is *not* $A \cup B$ – it is its own set.

$\{\, x \in U \mid x \text{ even} \,\} = \{2, 4, 6, 8\}$ **– the even members of the universe.**

Example 8

List every subset of $A \cap B$ and confirm the count matches 2^n.

SOLUTION

1. First find the set itself: $A \cap B = \{3, 4\}$, which has $n = 2$ elements. *overlap from Ex 2*
2. List subsets by size: the empty set $\varnothing$; the singletons $\{3\}, \{4\}$; and the whole set $\{3, 4\}$.
3. Count them: $1 + 2 + 1 = 4$. The formula predicts $2^n = 2^2 = 4$ – they agree. *each element: in or out*

The 4 subsets are $\varnothing, \{3\}, \{4\}, \{3, 4\}$, matching $2^2 = 4$.

Example 9

A survey of 40 people finds 22 like tea (T), 20 like coffee (C), 18 like juice (J); 10 like tea and coffee, 8 tea and juice, 9 coffee and juice, and 5 like all three. How many like at least one drink?

SOLUTION

1. Three-set inclusion–exclusion: add the singles, subtract each pair overlap, then add back the triple.
   +singles −pairs +triple
2. Add the singles: $22 + 20 + 18 = 60$.
3. Subtract the three pairwise overlaps: $60 - (10 + 8 + 9) = 60 - 27 = 33$. *pairs counted twice*
4. Add back the triple overlap (subtracted one time too many): $33 + 5 = 38$.

38 people like at least one drink (so $40 - 38 = 2$ like none).

Note: The count formula $|A \cup B| = |A| + |B| - |A \cap B|$ subtracts the overlap once because adding $|A|$ and $|B|$ counts every shared element **twice**. This "inclusion–exclusion" is the arithmetic behind the Venn picture; with three sets it extends to $|A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C|$.

⚠ Common Mistakes

- Confusing $\in$ with $\subseteq$. "$3 \in A$" (an *element*) is true for $A = \{1, 2, 3\}$, but "$3 \subseteq A$" is meaningless – only a *set* like $\{3\}$ can be a subset.
- Mixing up $\cup$ and $\cap$. Union $\cup$ (the wider "U") is **or** / merge; intersection $\cap$ is **and** / overlap. The overlap is never bigger than either set.
- Adding group sizes without removing the overlap: $|A \cup B| \neq |A| + |B|$ unless the sets are disjoint. The shared elements get counted twice.
- Forgetting the universe when taking a complement. $\complement A$ depends entirely on U – the complement of the evens is the odds only if U is the integers.
- Thinking $\varnothing$ is "nothing" so it cannot be a subset. In fact $\varnothing \subseteq B$ for **every** set B – there is no element of $\varnothing$ to violate the rule.

Quiz

23.1.Q1. With $A = \{1, 2, 3\}$ and $B = \{2, 3, 4\}$, what is $A \cap B$?

A. $\{1, 2, 3, 4\}$

B. $\{2, 3\}$

C. $\{1, 4\}$

D. $\varnothing$

23.1.Q2. With the same A, B, what is $A \cup B$?

23.1.Q3. Which symbol means "is an element of"?

A. $\subseteq$

B. $\in$

C. $\cup$

D. $\complement$

23.1.Q4. True or False: $\{1, 2\} \subseteq \{1, 2, 3\}$.

A. True

B. False

23.1.Q5. True or False: the empty set $\varnothing$ is a subset of every set.

A. True

B. False

23.1.Q6. Set intersection $\cap$ corresponds to which logic connective?

A. OR

B. AND

C. NOT

D. IMPLIES

23.1.Q7. If $U = \{1, 2, 3, 4, 5\}$ and $A = \{2, 4\}$, what is $\complement A$?

23.1.Q8. In a group, 20 like tea, 15 like coffee, 8 like both. How many like at least one?

23.1.Q9. True or False: $A \cap B$ can have more elements than A.

A. True

B. False

🐍 From Concept to Code (Python)

Python has a built-in `set` type whose operators line up one-to-one with the maths: `|` is union (∪), `&` is intersection (∩), `-` is difference (which gives a *relative* complement), and `in` tests membership (∈). Sets automatically drop duplicates, exactly like the mathematical definition.

```
A = {1, 2, 3, 4}
B = {3, 4, 5, 6}
print(A | B)      # union  ∪
print(A & B)      # intersection  ∩
print(A - B)      # in A but not B
print(3 in A)     # membership  ∈
```

```
output
{1, 2, 3, 4, 5, 6}
{3, 4}
{1, 2}
True
```

```
U = set(range(1, 9))        # universe {1..8}
A = {1, 2, 3, 4}
complement_A = U - A         # CA within U
print(complement_A)
print({1, 2} <= A)           # subset test  ⊆
```

```
output
{5, 6, 7, 8}
True
```

Try it:

1. Build the sets A=`{2,4,6,8}` and B=`{1,2,3,4}` and print `A & B` and `A | B`.
2. With a universe `U = set(range(10))`, compute the complement of `{0,2,4,6,8}` using `U - ...` and check it equals the odd numbers below 10.

Answer Key – Chapter 23 – Sets & Venn Diagrams

23.1.Q1. $\{2, 3\}$ – the elements in *both* sets.

23.1.Q2. $\{1, 2, 3, 4\}$ – everything in either set, duplicates listed once.

23.1.Q3. $\in$. ($\subseteq$ is "subset".)

23.1.Q4. True – every element of $\{1, 2\}$ is in $\{1, 2, 3\}$.

23.1.Q5. True – it has no element that could fail the subset rule.

23.1.Q6. AND – $x \in A \cap B$ means $x \in A$ **and** $x \in B$.

23.1.Q7. $\{1, 3, 5\}$ – the universe with A removed.

23.1.Q8. $27 - 20 + 15 - 8$ by inclusion–exclusion.

23.1.Q9. False – the overlap is a subset of A, so $|A \cap B| \leq |A|$.

Chapter 24 · Boolean Algebra

> *How to treat AND, OR and NOT as an **algebra** on the values 0 and 1 – using laws that look like ordinary algebra to simplify logical expressions and the switching circuits they describe.*

The Idea Behind It

Boolean algebra mirrors logic ($\cdot$ is AND, $+$ is OR, with $1 + 1 = 1$), and its laws are just truth-table facts, like absorption $A + A \cdot B = A$. *but why?*

Boolean algebra writes 1 for true and 0 for false, with $A \cdot B$ for AND and $A + B$ for OR, so a law is not something to memorise on faith, it is a claim that two expressions match on every input. Since each variable is only 0 or 1, "every input" is just a finite table, and a law is *true* precisely when its two columns agree. Take the **absorption law**, which says the B branch is redundant:

$$\boxed{A + A \cdot B = A}$$

There are only two cases for A, and they settle it without even splitting on B:

- **If $A = 0$:** then $A \cdot B = 0$, so $A + A \cdot B = 0 + 0 = 0 = A$.
- **If $A = 1$:** then $A + (\text{anything}) = 1$ (OR saturates, $1 + 1 = 1$), so $A + A \cdot B = 1 = A$.

The full table confirms both lines, the result column equals the A column exactly:

A	B	$A \cdot B$	$A + A \cdot B$	A
0	0	0	0	0
0	1	0	0	0
1	0	0	1	1
1	1	1	1	1

$$\therefore \; A + A \cdot B = A$$

Reading $\cdot$ as AND and $+$ as OR, this is the logic fact "$p \vee (p \wedge q) \equiv p$" in arithmetic dress, and in a circuit it means a whole switch can be **removed** without changing the output. Every Boolean law is an identity you can check this same way. *each law = a truth-table identity*

In ordinary algebra letters stand for numbers and you simplify with rules like $a + 0 = a$. **Boolean algebra** does the same thing for logic: the letters stand for **truth values**, written 1 for true and 0 for false, and a small set of laws lets you rewrite a complicated logical expression as a simpler one. This is not an academic game – every digital chip, every `if` condition, and every electrical switching circuit is a Boolean expression, and simplifying it means fewer gates, fewer mistakes and faster code.

BOOLEAN VALUES & OPERATIONS

A **Boolean variable** takes only two values: 1 (true) or 0 (false). There are three operations: - **AND**, written $A \cdot B$ or AB, is 1 only when *both* are 1. (This is $\wedge$.) - **OR**, written $A + B$, is 1 when *at least one* is 1. (This is $\vee$.) - **NOT**, written $\overline{A}$ (a bar) or A', flips the value. (This is $\neg$.)

The notation is deliberately arithmetic: AND behaves like **multiplication** and OR like **addition**, which is why $A \cdot B$ and $A + B$ are used. The defining tables (using $1/0$) are:

A	B	$A \cdot B$ (AND)	$A + B$ (OR)
1	1	1	1
1	0	0	1
0	1	0	1
0	0	0	0

One arithmetic warning right away: $1 + 1 = 1$ here, **not** 2. OR saturates at 1 because "true or true" is simply true; there is no value above 1. With that single exception, the multiplication intuition for AND is exact: $A \cdot B$ is 1 only when neither factor is 0.

The laws of Boolean algebra

Each law below holds for *every* value of the variables (check by trying 0 and 1). They come in **dual** pairs – swap $\cdot \leftrightarrow +$ and $0 \leftrightarrow 1$ to turn any law into its partner.

Law	AND form	OR form
Identity	$A \cdot 1 = A$	$A + 0 = A$
Null (domination)	$A \cdot 0 = 0$	$A + 1 = 1$
Idempotent	$A \cdot A = A$	$A + A = A$
Complement	$A \cdot \overline{A} = 0$	$A + \overline{A} = 1$
Commutative	$A \cdot B = B \cdot A$	$A + B = B + A$
Associative	$(AB)C = A(BC)$	$(A+B)+C = A+(B+C)$
Distributive	$A(B+C) = AB + AC$	$A + BC = (A+B)(A+C)$

Two more stand on their own. **Double negation** undoes a bar, and **De Morgan's laws** (from Chapter 21, now in Boolean dress) let a bar pass through a bracket while flipping the operation:

$$\overline{\overline{A}} = A$$

$$\overline{A \cdot B} = \overline{A} + \overline{B} \qquad \overline{A + B} = \overline{A} \cdot \overline{B}$$

Note: The OR-distributive law $A + BC = (A + B)(A + C)$ has **no counterpart in ordinary number algebra** – there, $a + bc \neq (a + b)(a + c)$. It works in Boolean algebra precisely because the values are only 0 and 1. This is the one law most likely to surprise you.

Boolean variable idempotent law duality

Switching circuits

Boolean algebra was born to describe **switching circuits**. Picture switches on a wire between a battery and a lamp; a switch passes current when its variable is 1:

- Switches **in series** (one after another) light the lamp only when *all* are closed – this is **AND** ($A \cdot B$).
- Switches **in parallel** (side-by-side branches) light the lamp when *any* one is closed – this is **OR** ($A + B$).

So a circuit *is* a Boolean expression, and simplifying the expression with the laws above means building the same behaviour with **fewer switches** (or, in a chip, fewer logic gates). That is exactly what the worked examples do.

Worked examples

Example 1

Simplify $A + A \cdot B$.

SOLUTION

1. Factor out the common A: $A + A \cdot B = A(1 + B)$. take out A
2. By the null law, $1 + B = 1$.
3. So the expression is $A \cdot 1$, and by identity $A \cdot 1 = A$.

$A + A \cdot B = A$ **(the "absorption" law – the B branch is redundant).**

Example 2

Simplify $A \cdot \overline{A} + B$.

SOLUTION

1. Apply the complement law to the first term: $A \cdot \overline{A} = 0$. a value vs its bar
2. The expression becomes $0 + B$.
3. By identity, $0 + B = B$.

$A \cdot \overline{A} + B = B$

Example 3

Use De Morgan to rewrite $\overline{A \cdot B}$ as a sum, then evaluate at $A = 1, B = 0$.

SOLUTION

1. De Morgan: $\overline{A \cdot B} = \overline{A} + \overline{B}$. bar product → OR
2. Substitute $A = 1,\ B = 0$: $\overline{1} + \overline{0} = 0 + 1$.
3. OR of 0 and 1 is 1.

$\overline{A \cdot B} = \overline{A} + \overline{B}$, **which is** 1 **at** $A = 1, B = 0$.

Example 4

Simplify $(A + B)(A + \overline{B})$.

SOLUTION

1. Expand with the OR-distributive (FOIL): $AA + A\overline{B} + BA + B\overline{B}$. multiply out
2. Idempotent: $AA = A$. Complement: $B\overline{B} = 0$.
3. Now $A + A\overline{B} + AB + 0 = A + A\overline{B} + AB$.
4. Factor A: $A(1 + \overline{B} + B) = A \cdot 1 = A$. bracket saturates to 1

$(A + B)(A + \overline{B}) = A$

Example 5

Prove the absorption law $A(A + B) = A$ by a value table.

SOLUTION

1. Tabulate all four input combinations and compute $A + B$, then $A(A + B)$. try every row
2. Row A=1, B=1: A+B=1, $A \cdot 1 = 1 = A$. Row A=1, B=0: A+B=1, gives $1 = A$.
3. Row A=0, B=1: A+B=1, $0 \cdot 1 = 0 = A$. Row A=0, B=0: $0 = A$.
4. In every row the result equals A. column matches A

$A(A + B) = A$ **for all inputs – confirmed.**

Example 6

A lamp is wired so it lights when (A AND B) OR (A AND NOT B). Simplify the circuit.

SOLUTION

1. Translate to Boolean: $A \cdot B + A \cdot \overline{B}$.
2. Factor out A: $A(B + \overline{B})$. take out A
3. Complement law: $B + \overline{B} = 1$.
4. Identity: $A \cdot 1 = A$.

The lamp depends on A alone – switch B can be removed entirely.

Example 7

Simplify $A\overline{B} + B$ (a disguised absorption).

SOLUTION

1. The term $A\overline{B}$ shares no obvious factor with B, so use the OR-distributive law in reverse on the *whole* expression.
2. Treat it as $\overline{B} \cdot A + B$. The law $X + \overline{X}Y = X + Y$ applies with $X = B, Y = A$: $B + \overline{B}A = B + A$.
   the bar term loses its bar
3. So the expression is just $A + B$.

$A\overline{B} + B = A + B$ **– the $\overline{B}$ drops away.**

Example 8

Use De Morgan, then simplify $\overline{A + B} + A$.

SOLUTION

1. Apply De Morgan to the bar over the sum: $\overline{A + B} = \overline{A} \cdot \overline{B}$. bar sum → AND of bars
2. The expression becomes $\overline{A}\,\overline{B} + A$.
3. Now $A + \overline{A}\,\overline{B}$ is the absorption form $X + \overline{X}Y = X + Y$ with $X = A,\ Y = \overline{B}$: $A + \overline{A}\,\overline{B} = A + \overline{B}$.
   drop the $\overline{A}$

$\overline{A + B} + A = A + \overline{B}$.

Example 9

Simplify the three-variable expression $ABC + AB\overline{C} + A\overline{B}$, then read off how many switches survive.

SOLUTION

1. Group the first two terms; they share AB: $ABC + AB\overline{C} = AB(C + \overline{C})$. *factor AB*
2. Complement law inside the bracket: $C + \overline{C} = 1$, so that group is $AB \cdot 1 = AB$.
3. The expression is now $AB + A\overline{B} = A(B + \overline{B}) = A \cdot 1$. *factor A, then saturate*
4. Identity gives A – every term collapses onto a single switch.

$ABC + AB\overline{C} + A\overline{B} = A$ **– switches B and C are both redundant.**

A	B	C	$ABC + AB\overline{C} + A\overline{B}$	A
1	1	1	1	1
1	1	0	1	1
1	0	1	1	1
1	0	0	1	1
0	1	1	0	0
0	0	0	0	0

Note: A simplification you can trust on paper you can trust in silicon. Every law here is just a truth-table identity; reducing $A \cdot B + A \cdot \overline{B}$ to A removes a real switch (or gate) while leaving the output unchanged on all inputs. The table above spot-checks Example 9 – the long expression and plain A agree in every row.

⚠ Common Mistakes

- Writing $1 + 1 = 2$. In Boolean algebra OR saturates: $1 + 1 = 1$. There is no value above 1.
- Mis-applying De Morgan by forgetting to flip the operation: $\overline{A \cdot B}$ is $\overline{A} + \overline{B}$ (a **sum**), **not** $\overline{A} \cdot \overline{B}$. The bar over a product becomes OR.
- Distributing a bar over a single term wrongly, e.g. claiming $\overline{A + B} = \overline{A} + \overline{B}$. The correct De Morgan result is $\overline{A} \cdot \overline{B}$.
- Assuming the OR-distributive $A + BC = (A + B)(A + C)$ "must be wrong" because it fails for numbers. It is a genuine Boolean law – values are only 0 and 1.
- Confusing series and parallel: switches **in series** are AND (all must close), switches **in parallel** are OR (any closes the circuit). Swapping them inverts the logic.

Quiz

24.1.Q1. In Boolean algebra, what is $1 + 1$?

A. 2

B. 0

C. 1

D. undefined

24.1.Q2. Simplify $A + 1$.

A. A

B. 1

C. 0

D. $\overline{A}$

24.1.Q3. Simplify $A \cdot \overline{A}$.

24.1.Q4. State De Morgan for $\overline{A + B}$.

24.1.Q5. True or False: switches connected in parallel implement AND.

A. True

B. False

24.1.Q6. Simplify $A + A \cdot B$.

A. A

B. B

C. $A \cdot B$

D. $A + B$

24.1.Q7. True or False: $A \cdot A = A$.

A. True

B. False

24.1.Q8. Simplify $A \cdot B + A \cdot \overline{B}$.

24.1.Q9. Which law lets you swap every $\cdot$ with $+$ and 0 with 1 to get another true law?

A. Identity

B. Duality

C. Idempotent

D. Associative

🐍 From Concept to Code (Python)

Python uses `and`/`or`/`not` for logic, but it also has **bitwise** operators `& | ~ ^` that act on the bits (0s and 1s) of integers – a direct preview of Boolean algebra in hardware. On the single bits 0 and 1, `&` is AND, `|` is OR, and `^` is XOR ("exclusive or": 1 when the two bits *differ*). Also handy: Python lets `True` and `False` behave as 1 and 0 in arithmetic.

```
print(1 & 1, 1 & 0)     # bitwise AND  (·)
print(1 | 0, 0 | 0)     # bitwise OR   (+)
print(1 ^ 1, 1 ^ 0)     # XOR: 1 only when bits differ
print(True + True)      # booleans as 1+1 in arithmetic
```

```
output
1 0
1 0
0 1
2
```

```
# absorption law  A + A·B == A  checked for all inputs
for A in (0, 1):
    for B in (0, 1):
        assert (A | (A & B)) == A
print("absorption law holds for every A, B")
```

```
output
absorption law holds for every A, B
```

Try it:

1. Print a truth table for XOR by looping `for A in (0,1): for B in (0,1):` and showing `A ^ B`.
2. Verify De Morgan in code: assert that `~(A & B) & 1 == ((~A & 1) | (~B & 1))` for every `A,B` in `(0,1)`. (The `& 1` keeps results to a single bit.)

Answer Key – Chapter 24 – Boolean Algebra

24.1.Q1. 1 – OR is true when at least one input is true, and it never exceeds 1.

24.1.Q2. 1 – the null (domination) law for OR.

24.1.Q3. 0 – the complement law: a value and its negation are never both true.

24.1.Q4. $\overline{A} \cdot \overline{B}$ – bar splits, OR becomes AND.

24.1.Q5. False – parallel switches implement **OR**; series switches implement AND.

24.1.Q6. A – the absorption law; the AB term is redundant.

24.1.Q7. True – the idempotent law for AND.

24.1.Q8. A – factor to $A(B + \overline{B}) = A \cdot 1 = A$.

24.1.Q9. Duality – every Boolean law has a dual partner.

Chapter 25 · Logical Reasoning & Validity

*How to lay an argument out as premises leading to a conclusion, tell a **valid** form from a **sound** one, apply modus ponens and modus tollens, and spot the two classic fallacies that fool almost everyone.*

The Idea Behind It

An argument can be valid yet have an untrue conclusion, because validity is about *form*, not *truth*.

but why?

Validity asks one hypothetical question of the *shape* of an argument: *if* every premise were true, would the conclusion be **forced**? **Soundness** asks more, that the form is valid **and** the premises are in fact true. So validity alone never promises a true conclusion: feed a valid form a false premise and the conclusion can be anything. "All cats can fly; Felix is a cat; therefore Felix flies" is airtight in form (valid) but built on a false premise (unsound). The governing test:

$$\boxed{\text{valid} \iff \text{no row makes every premise true while the conclusion is false}}$$

That test also exposes a famous **fallacy**, *affirming the consequent*: from $p \Rightarrow q$ and q, wrongly inferring p. Tabulate the premises $p \Rightarrow q$ and q against the conclusion p:

p	q	$p \Rightarrow q$	premise q	conclusion p
T	T	T	T	T
T	F	F	F	T
F	T	T	T	F
F	F	T	F	F

Look at the row $p = \mathsf{F},\ q = \mathsf{T}$: both premises are `true` ($p \Rightarrow q$ is T and q is T) yet the conclusion p is `false`. One such counterexample row is fatal.

$$\therefore \text{ affirming the consequent is } \boxed{\text{invalid}}$$

"If dog then mammal; it is a mammal; therefore a dog" fails on exactly this row, a cat. Validity is structure; soundness adds true premises; and a true-sounding conclusion never excuses a broken form.

form first, truth second

Everything in this Part has been building to one skill: deciding when a conclusion really *follows*. An **argument** is a set of starting claims (the **premises**) offered in support of a final claim (the **conclusion**). The central question of logic is not "are the premises true?" but "**if** they were true, would the conclusion have to be true as well?" Getting this distinction right is what separates careful reasoning from persuasive-sounding nonsense.

ARGUMENT, PREMISES, CONCLUSION

An **argument** consists of one or more **premises** (assumed statements) and a **conclusion** (the statement they are meant to establish). We display it with the premises stacked above a line and the conclusion below, the line standing for "**therefore**" ($\therefore$).

VALID VS SOUND

An argument is **valid** when its *form* guarantees the conclusion: it is impossible for all the premises to be true while the conclusion is false. An argument is **sound** when it is **both** valid **and** has premises that are actually true. Validity is about *structure*; soundness adds *truth*.

This split is the heart of the chapter. A **valid** argument can have a false conclusion – *if* one of its premises is false. A **true** conclusion can come from an **invalid** argument by sheer luck. Only a **sound** argument gives you both a trustworthy form and a guaranteed-true conclusion.

Note: Validity is hypothetical: it asks "what *if* the premises held?" An argument can be perfectly valid yet completely unsound. "All cats can fly; Felix is a cat; therefore Felix can fly" is **valid** (the form is airtight) but **unsound** (the first premise is false).

Two valid forms worth memorising

Most everyday deduction runs through a conditional $p \Rightarrow q$ ("if p then q", Chapter 20). Two forms built on it are always valid:

MODUS PONENS

From $p \Rightarrow q$ and p, conclude q. ("Affirm the antecedent.")

$$\frac{p \Rightarrow q, \quad p}{\therefore \; q}$$

If it rains the ground gets wet; it is raining; therefore the ground is wet.

MODUS TOLLENS

From $p \Rightarrow q$ and $\neg q$, conclude $\neg p$. ("Deny the consequent.")

$$\frac{p \Rightarrow q, \quad \neg q}{\therefore \; \neg p}$$

*If it rains the ground gets wet; the ground is **not** wet; therefore it is **not** raining.*

Two fallacies that imitate them

Each valid form has an evil twin – an **invalid** pattern that looks almost identical. These are the errors to police hardest:

Pattern	Premises	Drawn conclusion	Status
Modus ponens	$p \Rightarrow q$, and p	q	VALID
Modus tollens	$p \Rightarrow q$, and $\neg q$	$\neg p$	VALID
Affirming the consequent	$p \Rightarrow q$, and q	p	FALLACY
Denying the antecedent	$p \Rightarrow q$, and $\neg p$	$\neg q$	FALLACY

Why are the bottom two wrong? $p \Rightarrow q$ allows q to be true for *other* reasons. "If it rains, the ground is wet" does not promise rain is the **only** way the ground gets wet – a sprinkler would do it. So learning the ground is wet (affirming the consequent) does **not** let you conclude it rained, and learning it did not rain (denying the antecedent) does **not** let you conclude the ground is dry.

valid argument sound argument modus ponens modus tollens fallacy

Testing validity with a truth table

To check *any* argument: build a truth table of the premises and conclusion. The argument is **valid** exactly when **every row that makes all premises true also makes the conclusion true**. A single row with all-true premises and a false conclusion (a **counterexample row**) proves it invalid – the same counterexample idea from Chapter 22.

Worked examples

Example 1

Name the form and judge it: "If a number is divisible by 4, it is even. 12 is divisible by 4. Therefore 12 is even."

SOLUTION

1. Let p = "divisible by 4", q = "even". Premises: $p \Rightarrow q$ and p; conclusion q.
2. Affirming p then concluding q is **modus ponens** – valid. *affirm antecedent*
3. Both premises are actually true, so it is also sound. *valid + true premises*

Modus ponens – valid and sound.

Example 2

Judge: "If it is a square, it has four sides. This shape does **not** have four sides. Therefore it is **not** a square."

SOLUTION

1. p = "square", q = "four sides". Premises: $p \Rightarrow q$ and $\neg q$; conclusion $\neg p$.
2. Denying $\neg q$ to conclude $\neg p$ is **modus tollens**. *deny consequent*

Modus tollens – valid.

Example 3

Spot the flaw: "If I study, I pass. I passed. Therefore I studied."

SOLUTION

1. p = "study", q = "pass". Premises: $p \Rightarrow q$ and q; conclusion p.
2. Concluding p from the consequent q is **affirming the consequent**. *wrong direction*
3. Invalid: I might have passed for another reason (an easy exam), so q does not force p.

Affirming the consequent – a fallacy (invalid).

Example 4

Spot the flaw: "If it is a dog, it is a mammal. It is not a dog. Therefore it is not a mammal."

SOLUTION

1. p = "dog", q = "mammal". Premises: $p \Rightarrow q$ and $\neg p$; conclusion $\neg q$.
2. Concluding $\neg q$ from $\neg p$ is **denying the antecedent**. *not the only way*
3. Invalid: a cat is not a dog yet is still a mammal – a counterexample.

Denying the antecedent – a fallacy (invalid).

Example 5

Show by truth table that affirming the consequent is invalid.

SOLUTION

1. Tabulate p, q with premises $p \Rightarrow q$ and q, and conclusion p.
2. Hunt for a row where both premises are true but the conclusion is false. *counterexample row*
3. Row $p = F,\ q = T$: $p \Rightarrow q$ is T, premise q is T – both true – yet conclusion p is F.
4. That one row settles it.

The row $p = F,\ q = T$ makes the premises true and the conclusion false, so the form is invalid.

p	q	$p \Rightarrow q$	premise q	conclusion p
T	T	T	T	T
T	F	F	F	T
F	T	T	T	F
F	F	T	F	F

Example 6

Problem solving – chain the rules. Premises: (1) "If the alarm rings, there is smoke." (2) "If there is smoke, there is fire." (3) "There is **no** fire." What follows about the alarm?

SOLUTION

1. Let a=alarm rings, s=smoke, f=fire. Premises: $a \Rightarrow s$, $s \Rightarrow f$, $\neg f$.
2. From $s \Rightarrow f$ and $\neg f$, modus tollens gives $\neg s$ (no smoke). *tollens #1*
3. From $a \Rightarrow s$ and $\neg s$, modus tollens again gives $\neg a$. *tollens #2*
4. Each step is a valid form, so the chain is valid.

The alarm did not ring ($\neg a$) – by two applications of modus tollens.

Example 7

Name the valid form: "If it freezes, the pipes burst. If the pipes burst, the basement floods. Therefore if it freezes, the basement floods."

SOLUTION

1. Let p=freezes, q=pipes burst, r=basement floods. Premises: $p \Rightarrow q$ and $q \Rightarrow r$; conclusion $p \Rightarrow r$.
2. The shared middle term q links the two conditionals: the consequent of the first is the antecedent of the second. *link through* q
3. Discharge q to chain the ends: this is the **hypothetical syllogism**, and it is valid.

Hypothetical syllogism – $p \Rightarrow r$ is valid (conditionals chain transitively).

Example 8

Judge: "Either the server is down or the cable is loose. The server is **not** down. Therefore the cable is loose."

SOLUTION

1. Let d=server down, c=cable loose. Premises: $d \vee c$ and $\neg d$; conclusion c.
2. An "or" guarantees *at least one* disjunct holds. Knocking out d with $\neg d$ leaves the other standing. *eliminate* d
3. Concluding c is the **disjunctive syllogism** – valid.

Disjunctive syllogism – c follows validly.

Example 9

Problem solving with mixed rules. Premises: (1) $p \vee q$. (2) $p \Rightarrow r$. (3) $\neg r$. (4) $q \Rightarrow s$. What can you conclude about s ?

SOLUTION

1. Start where a negation bites: from $p \Rightarrow r$ and $\neg r$, modus tollens gives $\neg p$. *tollens*
2. Feed that into the disjunction $p \vee q$: with $\neg p$, disjunctive syllogism forces q. *eliminate p*
3. Now $q \Rightarrow s$ with the established q gives s by modus ponens. *ponens*
4. Three valid steps in sequence, so the whole deduction is valid.

Therefore s – via modus tollens, then disjunctive syllogism, then modus ponens.

Note: Real problem solving strings these forms together. Label each statement with a letter, write the premises symbolically, then apply modus ponens / modus tollens / disjunctive syllogism one valid step at a time until you reach (or rule out) the conclusion. Naming the form at each step is your guarantee the chain stays valid.

⚠ Common Mistakes

- Confusing valid with true. A **valid** argument can have a false conclusion if a premise is false; **soundness** (valid + true premises) is what guarantees a true conclusion.
- Affirming the consequent: from $p \Rightarrow q$ and q, wrongly concluding p. The consequent q may hold for other reasons, so it does not force p.
- Denying the antecedent: from $p \Rightarrow q$ and $\neg p$, wrongly concluding $\neg q$. "p is one way to get q" does not make it the *only* way.
- Treating $p \Rightarrow q$ as if it also claimed $q \Rightarrow p$. An implication is one-directional; assuming the converse is exactly what powers both fallacies.
- Thinking a true-sounding conclusion proves the argument good. The conclusion can be true by accident even when the reasoning is invalid – always check the *form*.

Quiz

25.1.Q1. From "$p \Rightarrow q$" and "p", what validly follows?

A. $\neg q$

B. q

C. $\neg p$

D. nothing

25.1.Q2. From "$p \Rightarrow q$" and "$\neg q$", what validly follows?

A. p

B. q

C. $\neg p$

D. $\neg q$ only

25.1.Q3. What is the name of concluding p from "$p \Rightarrow q$" and "q"?

25.1.Q4. True or False: an argument can be valid but not sound.

A. True

B. False

25.1.Q5. True or False: a sound argument is always valid.

A. True

B. False

25.1.Q6. Is this valid? "If it is Tuesday, the shop is open. It is not Tuesday. Therefore the shop is closed."

A. Valid

B. Invalid

25.1.Q7. What must a truth-table row show to *disprove* an argument's validity?

25.1.Q8. Premises: $a \Rightarrow b$ and $b \Rightarrow c$ and a. What follows about c?

25.1.Q9. True or False: validity depends on whether the premises are actually true.

A. True

B. False

🐍 From Concept to Code (Python)

You can encode a logical rule as a Python **function** and then use `assert` to *check* that an argument form behaves as claimed for every truth assignment. `assert` raises an error if its condition is ever `False`, so an assert that survives a loop over all cases is a tiny machine-checked proof. We model $p \Rightarrow q$ as `(not p) or q`.

```
def implies(p, q):
    return (not p) or q          # p ⇒ q

# modus ponens is valid: whenever (p⇒q) and p hold, q holds.
for p in (False, True):
    for q in (False, True):
        if implies(p, q) and p:      # both premises true
            assert q                 # conclusion must hold
print("modus ponens is valid")
```

```
output
modus ponens is valid
```

```
# affirming the consequent is INVALID: find the counterexample.
def implies(p, q): return (not p) or q
for p in (False, True):
    for q in (False, True):
        if implies(p, q) and q and not p:    # premises true, conclusion p
false
            print("counterexample: p =", p, " q =", q)
```

```
output
counterexample: p = False  q = True
```

Try it:

1. Write `implies()` and assert that **modus tollens** is valid: whenever `implies(p,q)` and `not q` hold, `not p` follows, for all `p,q`.
2. Encode the rule "a year is a leap year if it is divisible by 4" as `def leap(y): return y % 4 == 0`, then `assert leap(2024)` and `assert not leap(2023)`.

Answer Key – Chapter 25 – Logical Reasoning & Validity

25.1.Q1. q – this is modus ponens (valid).

25.1.Q2. $\neg p$ – this is modus tollens (valid).

25.1.Q3. Affirming the consequent – an invalid form (fallacy).

25.1.Q4. True – valid form with a false premise is valid yet unsound.

25.1.Q5. True – "sound" is defined as valid **plus** true premises.

25.1.Q6. Invalid – denying the antecedent; the shop could open on other days too.

25.1.Q7. A row where every premise is true but the conclusion is false (a counterexample row).

25.1.Q8. c – two modus ponens steps: a gives b, then b gives c.

25.1.Q9. False – validity is about form only; truth of premises is the separate question of soundness.



PART VI

Algebra: Equations, Inequalities & Functions

From arithmetic to algebra: using letters for numbers, solving equations and inequalities step by step, and meeting functions and the coordinate plane.

Chapter 26 – Variables & Expressions · Chapter 27 – Solving Linear Equations · Chapter 28 – Inequalities · Chapter 29 – Functions & the Coordinate Plane

Chapter 26 · Variables & Expressions

How to read and use a variable as a stand-in for a number, build expressions from words, evaluate them by substitution, and simplify by collecting like terms.

The Idea Behind It

Why like terms add but unlike terms do not: $3x + 2x = 5x$, yet $3x + 2y$ is stuck. *but why?*

"Collecting like terms" is not a new rule to memorise: it is the **distributive law** $ab + ac = a(b + c)$ from Chapter 9, read **backwards** to pull out a shared factor. The terms $3x$ and $2x$ both contain the very same piece, x, so x can be factored out:

$$\boxed{ab + ac = a(b + c)}$$

$$3x + 2x = 3\ x + 2\ x = (3 + 2)\, x = 5x$$

The whole job was adding the **coefficients** 3 and 2 while the shared x came along for the ride. You can check it against a number: at $x = 4$, the left side is $12 + 8 = 20$ and $5x = 5 \times 4 = 20$ too. Now try the same move on $3x + 2y$. There is **no common letter** to take outside a bracket:

$$3x + 2y = ? \quad \text{no shared factor, so it stays } 3x + 2y$$

$$\therefore\ 3x + 2x = 5x, \qquad 3x + 2y \text{ will not combine}$$

So terms combine **exactly when** they share an identical variable part to factor out, and not otherwise. That single requirement is the whole idea of a "like term". *same variable part = factorable*

So far every quantity has been a fixed number. Algebra adds one powerful idea: a **letter that stands for a number we do not yet know, or one that can change**. Writing "a number plus 5" as $n + 5$ lets us describe a rule once and apply it to *any* value of n. That small move – from specific numbers to a symbol for "any number" – is what turns arithmetic into algebra.

VARIABLE

A letter (such as x, n, or t) used to represent a number. It may stand for an **unknown** value we want to find, or a **changing** value, like the t in "distance $= 60t$" where t is however many hours you have driven.

CONSTANT

A fixed number that does not change, such as the 5 in $n + 5$ or the 3 in $3x + 7$.

The anatomy of an expression

> **ALGEBRAIC EXPRESSION**
>
> A combination of numbers, variables, and operations – with **no equals sign**. $3x + 7$, $2(a - 4)$, and $\frac{n}{2} + 1$ are expressions. (Put an $=$ between two expressions and you get an *equation*, the subject of Chapter 27.)

An expression is made of **terms** separated by $+$ or $-$ signs. In $3x + 7$ there are two terms: $3x$ and 7.

> **TERM**
>
> A single number, variable, or product of them – the pieces an expression is added or subtracted from. In $5y - 2x + 9$, the terms are $5y$, $-2x$, and 9.

> **COEFFICIENT**
>
> The number multiplying a variable in a term. In $3x$ the coefficient is 3. A lone x has coefficient 1 (we just do not write it), and $-x$ has coefficient -1.

Two shorthand conventions you must know:

- Multiplication is written **without a sign**: $3x$ means $3 \times x$, and ab means $a \times b$.
- A variable times itself uses a **power**: $x \cdot x = x^2$.

Evaluating an expression

To **evaluate** an expression, **substitute** a number for each variable and then follow the order of operations (Chapter 1). The same expression gives different values for different inputs.

Simplifying – collecting like terms

> **LIKE TERMS**
>
> Terms with **exactly the same variable part** (same letters to the same powers). $4x$ and $7x$ are like terms; $4x$ and $4x^2$ are **not**, and $4x$ and $4y$ are not. Only like terms can be combined – add their coefficients and keep the variable part.

$$4x + 7x = 11x \qquad \text{but} \qquad 4x + 7y \text{ cannot be combined}$$

Worked examples

Example 1

Write an expression for "7 more than twice a number n".

SOLUTION

1. "Twice a number" is $2 \times n = 2n$. × = no sign
2. "7 more than" that means add 7: $2n + 7$.

$2n + 7$.

Example 2

Evaluate $3x + 7$ when $x = 5$.

SOLUTION

1. Substitute 5 for x: $3 \times 5 + 7$. swap in 5
2. Order of operations – multiply first: $3 \times 5 = 15$, giving $15 + 7$.
3. Then add: $15 + 7 = 22$.

$3x + 7 = 22$ **when** $x = 5$.

Example 3

Evaluate $2a - b$ when $a = 10$ and $b = 3$.

SOLUTION

1. Substitute both values: $2 \times 10 - 3$.
2. Multiply first: $2 \times 10 = 20$, giving $20 - 3$. × before −
3. Subtract: $20 - 3 = 17$.

$2a - b = 17$.

Example 4

Identify the terms and coefficients in $5y - 2x + 9$.

SOLUTION

1. Split at the $+$ and $-$ signs, keeping each sign with its term: $5y$, $-2x$, 9. sign stays
2. The coefficient of y is 5; the coefficient of x is -2.
3. The 9 is a **constant** term – it has no variable.

Terms: $5y$, $-2x$, 9; **coefficients:** 5 **and** -2; **constant:** 9.

Example 5

Simplify $4x + 3 + 2x + 5$ by collecting like terms.

SOLUTION

1. Group like terms: the x-terms $4x$ and $2x$; the constants 3 and 5.
2. Add coefficients of the x-terms: $4x + 2x = 6x$. 4+2=6
3. Add the constants: $3 + 5 = 8$.

$4x + 3 + 2x + 5 = 6x + 8$.

Example 6

Simplify $7a + 2b - 3a + 5b - 4$.

SOLUTION

1. Collect the a-terms: $7a - 3a = 4a$. 7−3=4
2. Collect the b-terms: $2b + 5b = 7b$.
3. The -4 has no like term, so it stays. Note $4a$ and $7b$ are not like terms and cannot merge.

$7a + 2b - 3a + 5b - 4 = 4a + 7b - 4$.

Example 7

Expand then simplify $3(x + 4) + 2x$.

SOLUTION

1. Distribute the 3 across the bracket (Chapter 9): $3 \times x + 3 \times 4 = 3x + 12$. multiply both
2. Rewrite the whole expression: $3x + 12 + 2x$.
3. Collect like terms: $3x + 2x = 5x$, then keep $+12$.

$3(x + 4) + 2x = 5x + 12$.

Example 8

Expand and simplify $2(3x - 1) - (x - 4)$ (mind the subtracted bracket).

SOLUTION

1. Distribute the first bracket: $2 \times 3x - 2 \times 1 = 6x - 2$. multiply both
2. The minus in front of the second bracket is -1, so it flips *every* inside sign: $-(x - 4) = -x + 4$.

   minus flips both
3. Gather: $6x - 2 - x + 4$. Collect x-terms: $6x - x = 5x$; constants: $-2 + 4 = 2$.

$2(3x - 1) - (x - 4) = 5x + 2$.

Example 9

Evaluate $\dfrac{a^2 - b}{c}$ when $a = 5$, $b = 7$ and $c = 3$.

SOLUTION

1. Substitute all three values: $\dfrac{5^{\,2} - 7}{3}$. *swap in the numbers*
2. The fraction bar groups the top, so finish it first. Power before subtraction: $5^2 = 25$, then $25 - 7 = 18$.
3. Now divide by the bottom: $18 \div 3 = 6$.

$\dfrac{a^2 - b}{c} = 6$ **when** $a = 5,\ b = 7,\ c = 3$.

Example 10

A taxi charges a 3 dollar flat fee plus 2 dollars per mile. Write an expression for the cost of an m-mile trip, then find the cost of a 7-mile trip.

SOLUTION

1. "2 dollars per mile" scales with distance: $2 \times m = 2m$ dollars. *rate × miles*
2. Add the flat fee: cost $= 2m + 3$ dollars.
3. Evaluate at $m = 7$: $2 \times 7 + 3 = 14 + 3 = 17$. *× before +*

Cost expression: $2m + 3$ dollars; a 7-mile trip costs 17 dollars.

Note: A variable always represents the **same** value everywhere it appears in a single expression: in $x + x + x$, all three xs are the same number, so the expression simplifies to $3x$. Substituting $x = 4$ gives 12 either way.

variable constant algebraic expression term coefficient like terms

Mental-Math Shortcut

Evaluate before you expand. If you only need the *value* of an expression for a given input, substitute the number first instead of simplifying the algebra – it is usually faster.

For $3(x + 4) + 2x$ at $x = 6$: read it straight as $3(6 + 4) + 2 \times 6 = 3 \times 10 + 12 = 42$. No need to expand to $5x + 12$ first (though that also gives $5 \times 6 + 12 = 42$).

Coefficient bookkeeping. When collecting like terms, just track the running coefficient as a signed sum. For $7a - 3a + a$, add the coefficients in your head: $7 - 3 + 1 = 5$, so the answer is $5a$. The variable part comes along for the ride.

⚠ Common Mistakes

- Combining unlike terms: $4x + 3 = 7x$. Wrong – 3 has no x, so it stays separate: $4x + 3$.
- Treating x^2 and x as like terms: $x^2 + x = x^3$ or $2x^2$. Both wrong – different powers, so $x^2 + x$ cannot be combined.
- Forgetting the invisible coefficient 1: simplifying $5x - x$ to 5 or $5x$. The x is $1x$, so $5x - x = 4x$.
- Dropping a sign when collecting: in $7a - 3a$, the second term is $-3a$, giving $4a$, not $10a$.
- Reading $3n$ as "3 then n" (like 35) instead of $3 \times n$. Juxtaposition means multiplication.

📝 Quiz

26.1.Q1. Write an expression for "5 less than a number x".

A. $5 - x$

B. $x - 5$

C. $5x$

D. $x + 5$

26.1.Q2. Evaluate $4n + 1$ when $n = 6$.

26.1.Q3. Evaluate $a^2 - 2b$ when $a = 5$ and $b = 4$.

26.1.Q4. What is the coefficient of x in the term $-x$?

A. 0

B. 1

C. -1

D. x

26.1.Q5. Simplify $6x + 2 - 4x + 9$.

26.1.Q6. Simplify $5a + 3b - 2a + b$.

26.1.Q7. True or False: $3x$ and $3y$ are like terms.

A. True

B. False

26.1.Q8. Expand and simplify $2(x + 5) + 3x$.

🐍 From Concept to Code (Python)

A Python **variable** works just like an algebraic one – a name holding a value – created with the assignment operator `=`. Python is *dynamically typed*: the same name can hold an `int`, then a `float`, then a `str`, and `type()` reports which. An **f-string** (an `f` before the quotes) drops a variable's value straight into text.

```
x = 5                  # assignment: the name x now holds 5
result = 3 * x + 7     # evaluate the expression 3x + 7
print(result)
print(f"When x = {x}, 3x + 7 = {result}")
```

```
output
22
When x = 5, 3x + 7 = 22
```

```
n = 6                  # int
print(type(n))
n = 6.0                # same name, now a float
print(type(n))
n = "six"              # now a string
print(type(n))
```

```
output
<class 'int'>
<class 'float'>
<class 'str'>
```

Try it:

1. Set `a = 10` and `b = 3`, then print the value of `2*a - b` using an f-string that shows both inputs.
2. Assign `price = 20`, then reassign `price = price * 0.9` (a 10% discount) and print the new value. What does reusing the name on both sides do?

Answer Key – Chapter 26 – Variables & Expressions

26.1.Q1. $x - 5$ – "less than" subtracts from the number, so the order is x minus 5.

26.1.Q2. $25 - 4 \times 6 = 24$, then $+1 = 25$.

26.1.Q3. $17 - 5^2 = 25$, $2 \times 4 = 8$, then $25 - 8 = 17$.

26.1.Q4. -1 – a bare $-x$ means $-1 \times x$.

26.1.Q5. $2x + 11 - 6x - 4x = 2x$ and $2 + 9 = 11$.

26.1.Q6. $3a + 4b - 5a - 2a = 3a$ and $3b + b = 4b$; the two cannot merge.

26.1.Q7. False – different variables, so they are not like terms.

26.1.Q8. $5x + 10$ – distribute to $2x + 10$, then $2x + 3x = 5x$.

Chapter 27 · Solving Linear Equations

How to solve a linear equation step by step – one-step, two-step, variables on both sides and fractional coefficients – by keeping it balanced, and how to check your answer.

The Idea Behind It

Why you are allowed to do the same thing to both sides and the answer survives. *but why?*

Every step of solving leans on one tiny fact about the word **equal**: if two quantities are truly the same, then changing both by the **same** amount leaves them still the same. Equals stay equal:

$$\boxed{\text{if } a = b,\ \text{then } a + c = b + c \text{ and } ac = bc}$$

Picture a balanced scale reading $a = b$. Drop the same weight c on **both** pans and it still balances; double **both** pans and it still balances. So each legal move turns a true equation into another true equation, with the **same** unknown satisfying it. Watch one step of $x + 7 = 12$, subtracting 7 from each side:

$$x + 7 = 12 \implies x + 7 \,{-7} = 12 \,{-7} \implies x = 5$$

Nothing was lost, because the move is **reversible**: adding 7 back to $x = 5$ returns the original line. A value works in one equation exactly when it works in the other, so $x = 5$ is the answer. Checking confirms it:

$$5 + 7 = 12 \ \checkmark$$

$$\therefore \ \text{same operation on both sides} \ \Rightarrow \ \text{same solution}, \quad x = 5$$

That is the entire engine of equation-solving: keep the scale balanced and peel the unknown free, one matched operation at a time. (The one care: never multiply or divide both sides by 0, which would erase the balance.) *balance in, balance out*

An **expression** like $3x + 2$ is just a recipe: it has no $=$ sign and nothing to *solve* – you can only simplify or evaluate it. An **equation** like $3x + 2 = 11$ makes a *claim*: that the two sides are equal. **Solving** an equation means finding the value of the unknown that makes that claim true. For a **linear** equation the unknown appears only to the first power (no x^2, no $\sqrt{x}$), and there is exactly one such value.

THE BALANCE PRINCIPLE

An equation is like a balanced scale. **Whatever you do to one side you must do to the other**, and the equation stays true. You may add, subtract, multiply or divide both sides by the same number (never dividing by zero), and the solution is unchanged.

To isolate the variable, peel away the operations attached to it using **inverse operations** – addition undoes subtraction, and multiplication undoes division:

$$+ \longleftrightarrow - \qquad\qquad \times \longleftrightarrow \div$$

Undo them in the **reverse** of the order-of-operations order: clear the $+$ and $-$ first, then the $\times$ and $\div$. The goal each time is to end with the variable **alone** on one side.

A reliable game plan

1. If there are fractions, **multiply through** by a common denominator to clear them.
2. **Collect** the variable terms on one side and the plain numbers on the other.
3. **Undo** addition/subtraction, then multiplication/division.
4. **Check** by substituting your answer back into the *original* equation.

Worked examples

Example 1

Solve the one-step equation $x + 7 = 12$.

SOLUTION

1. The variable has 7 added to it; the inverse of $+7$ is -7. undo +7
2. Subtract 7 from **both** sides: $x + 7 \ -7 \ = 12 \ -7$.
3. Simplify: $x = 5$.
4. Check: $5 + 7 = 12$ ✓.

$x = 5$

Example 2

Solve the one-step equation $\frac{x}{4} = 3$.

SOLUTION

1. The variable is divided by 4; the inverse is multiplying by 4.
2. Multiply both sides by 4: $\frac{x}{4} \times 4 = 3 \times 4$. ×4 both sides
3. Simplify: $x = 12$.
4. Check: $\frac{12}{4} = 3$ ✓.

$x = 12$

Example 3

Solve the two-step equation $3x + 2 = 11$.

SOLUTION

1. Undo the addition first: subtract 2 from both sides, giving $3x = 9$. −2 both sides
2. Undo the multiplication: $\div 3$ both sides, giving $x = 3$.
3. Check: $3(3) + 2 = 9 + 2 = 11$ ✓.

$x = 3$

Example 4

Solve $5 - 2x = 13$.

SOLUTION

1. Subtract 5 from both sides: $-2x = 8$.
2. Divide both sides by -2 (watch the sign): $x = \frac{8}{-2} = -4$. ÷ -2
3. Check: $5 - 2(-4) = 5 + 8 = 13$ ✓.

$x = -4$

Example 5

Solve with variables on both sides: $4x + 3 = x + 18$.

SOLUTION

1. Collect the variable terms on the left by subtracting x from both sides: $3x + 3 = 18$. -x both sides
2. Collect the numbers on the right by subtracting 3: $3x = 15$.
3. Divide both sides by 3: $x = 5$.
4. Check: left $= 4(5) + 3 = 23$; right $= 5 + 18 = 23$ ✓.

$x = 5$

Example 6

Solve with brackets: $2(x - 3) = 4x + 2$.

SOLUTION

1. Expand the bracket using the distributive property: $2x - 6 = 4x + 2$. distribute the 2
2. Subtract $2x$ from both sides to gather the variable on the right: $-6 = 2x + 2$.
3. Subtract 2: $-8 = 2x$. Divide by 2: $x = -4$.
4. Check: left $= 2(-4 - 3) = 2(-7) = -14$; right $= 4(-4) + 2 = -14$ ✓.

$x = -4$

Example 7

Solve the fractional equation $\frac{x}{3} + \frac{1}{2} = \frac{5}{6}$.

SOLUTION

1. Clear fractions by multiplying every term by the common denominator 6: $6 \cdot \frac{x}{3} + 6 \cdot \frac{1}{2} = 6 \cdot \frac{5}{6}$.

   ×6 everywhere
2. This gives $2x + 3 = 5$.
3. Subtract 3: $2x = 2$. Divide by 2: $x = 1$.
4. Check: $\frac{1}{3} + \frac{1}{2} = \frac{2}{6} + \frac{3}{6} = \frac{5}{6}$ ✓.

$x = 1$

Example 8

Solve with brackets on both sides: $3(x + 2) = 2(x - 1) + 9$.

SOLUTION

1. Expand each bracket. Left: $3 \times x + 3 \times 2 = 3x + 6$. Right: $2 \times x - 2 \times 1 + 9 = 2x + 7$.

   distribute both
2. The equation is now $3x + 6 = 2x + 7$. Subtract $2x$ from both sides: $x + 6 = 7$.
3. Subtract 6 from both sides: $x = 1$.
4. Check: left $= 3(1 + 2) = 9$; right $= 2(1 - 1) + 9 = 9$ ✓.

$x = 1$

Example 9

Solve the fractional equation with the variable on both sides: $\frac{2x}{3} - 1 = \frac{x}{2}$.

SOLUTION

1. Clear fractions with the common denominator 6, multiplying *every* term: $6 \cdot \frac{2x}{3} - 6 \cdot 1 = 6 \cdot \frac{x}{2}$.

   ×6 every term
2. This gives $4x - 6 = 3x$.
3. Subtract $3x$ from both sides to gather the variable: $x - 6 = 0$.
4. Add 6: $x = 6$.
5. Check: $\frac{2(6)}{3} - 1 = 4 - 1 = 3$ and $\frac{6}{2} = 3$ ✓.

$x = 6$

Example 10

A number is multiplied by 4, then 5 is subtracted; the result equals the number plus 7. Find the number.

SOLUTION

1. Let the number be n. Translate phrase by phrase: "multiplied by 4 then 5 subtracted" is $4n - 5$; "the number plus 7" is $n + 7$. words → symbols
2. Set them equal: $4n - 5 = n + 7$. Subtract n from both sides: $3n - 5 = 7$.
3. Add 5: $3n = 12$. Divide by 3: $n = 4$.
4. Check: $4(4) - 5 = 11$ and $4 + 7 = 11$ ✓.

The number is 4.

Note: Always substitute back into the **original** equation, not a line you rewrote – that way a slip in your working cannot hide. If both sides come out equal, your solution is correct.

Tip: When you divide both sides by a negative number, every sign flips. Slow down on that step: it is where most sign errors creep in.

equation expression inverse operation solution linear equation

⚡ Mental-Math Shortcut

For a clean two-step equation $ax + b = c$, you can read the answer off in one mental sweep: **subtract the constant, then divide by the coefficient**, i.e. $x = \dfrac{c - b}{a}$.

For $3x + 2 = 11$: $11 - 2 = 9$, then $9 \div 3 = 3$. Done.

A fast plausibility check without fully solving: substitute a **round guess** and see which way it misses. In $4x + 3 = 23$, trying $x = 5$ gives 23 exactly – so 5 is the answer. If a guess overshoots, the true value is smaller; if it undershoots, it is larger. This "nudge toward the target" instinct catches wrong answers before you commit to them.

⚠ Common Mistakes

- Changing only one side of the equation – e.g. subtracting 2 from the left of $3x + 2 = 11$ but not the right. The balance principle requires doing it to **both** sides.
- Forgetting to flip signs when dividing by a negative: from $-2x = 8$ the answer is $x = -4$, not $x = 4$.
- Distributing incorrectly: $2(x - 3)$ is $2x - 6$, **not** $2x - 3$. The factor multiplies every term inside the bracket.
- When clearing fractions, multiplying only the fraction terms and skipping the whole-number terms. **Every** term must be multiplied by the common denominator.
- Treating an expression like an equation – "solving" $3x + 2$. With no $=$ sign there is nothing to solve; you can only simplify or evaluate it.

Quiz

27.1.Q1. Solve $x - 9 = 4$.

27.1.Q2. Solve $6x = 42$.

A. $x = 6$

B. $x = 7$

C. $x = 36$

D. $x = 48$

27.1.Q3. Solve $2x + 5 = 17$.

27.1.Q4. Solve $\frac{x}{5} - 1 = 2$.

27.1.Q5. Solve $7 - 3x = 1$.

A. $x = -2$

B. $x = 2$

C. $x = \frac{8}{3}$

D. $x = 6$

27.1.Q6. Solve $5x - 4 = 2x + 11$.

27.1.Q7. Solve $3(x + 2) = 18$.

27.1.Q8. Is $x = 4$ a solution of $5x - 3 = 17$?

A. Yes

B. No

27.1.Q9. True or False: $4x + 1$ is an equation.

A. True

B. False

🐍 From Concept to Code (Python)

Python uses **=** to *assign* a value to a variable, but **==** to *test* whether two things are equal – the same distinction as an expression versus an equation. That makes == perfect for **checking a solution**: substitute your answer and let Python confirm the two sides match.

```
x = 3                      # '=' assigns the value 3 to x
print(3 * x + 2)           # evaluate the left side
print(3 * x + 2 == 11)     # '==' checks the equation: is it 11?
```

```
output
11
True
```

```
# Verify x = −4 solves  5 − 2x = 13
x = −4
left = 5 − 2 * x
print(left)
print(left == 13)          # True confirms the solution
```

```
output
13
True
```

Try it:

1. Set `x = 5` and check that `4*x + 3 == x + 18` is `True` (Example 5).
2. Pick a *wrong* value like `x = 2` for `3*x + 2 == 11` and confirm Python returns `False`.

Answer Key – Chapter 27 – Solving Linear Equations

27.1.Q1. $x = 13$ – add 9 to both sides.

27.1.Q2. $x = 7$ – divide both sides by 6.

27.1.Q3. $x = 6$ – subtract 5 ($2x = 12$), then divide by 2.

27.1.Q4. $x = 15$ – add 1 ($\frac{x}{5} = 3$), then multiply by 5.

27.1.Q5. $x = 2$ – subtract 7 ($-3x = -6$), divide by -3.

27.1.Q6. $x = 5$ – subtract $2x$ ($3x - 4 = 11$), add 4, divide by 3.

27.1.Q7. $x = 4$ – expand to $3x + 6 = 18$, or divide by 3 first to get $x + 2 = 6$.

27.1.Q8. Yes – $5(4) - 3 = 17$ ✓.

27.1.Q9. False – it has no $=$ sign, so it is an expression.

Chapter 28 · Inequalities

How to solve an inequality the same way you solve an equation – remembering to flip the sign when you multiply or divide by a negative – then show every answer on a number line.

The Idea Behind It

Why multiplying or dividing by a negative flips the sign: $2 < 3$ becomes $-2 > -3$. *but why?*

The flip is not an arbitrary "remember to do it" rule, it follows from what $a < b$ actually **means**: that the gap $b - a$ is **positive**. The governing fact is how multiplying by a negative changes a sign:

$$\boxed{a < b \iff b - a > 0, \qquad \text{and } (-1) \times (\text{positive}) = \text{negative}}$$

Start from a true statement, say $2 < 3$, so the gap $3 - 2 = 1$ is positive. Multiply that gap by -1: a positive becomes a **negative**, which reverses which number is bigger:

$$3 - 2 > 0 \xrightarrow{\times(-1)} -(3-2) = 2 - 3 < 0 \implies -3 < -2$$

Reading that last line the other way round, $-2 > -3$: the "$<$" has turned into "$>$". On the number line, negating **reflects every point across zero**, and a mirror reverses left-to-right order, so the smaller number ends up on the right. Hence the rule for solving:

$$\therefore\ a < b \implies -a > -b \qquad (\text{multiply by a negative} \Rightarrow \text{flip})$$

So when an inequality's final step is dividing by a negative, the sign must turn around to keep the statement true. Adding or subtracting never triggers it: only a negative **multiplier or divisor** does.

negate = reflect = reverse order

An equation such as $x = 5$ has exactly one answer. But many real questions have *many* answers at once: "I have at most £20 to spend", "you must be **at least** 18 to vote", "the temperature stayed **below** zero all day". To capture a whole range of values in one short statement we use an **inequality**.

INEQUALITY

A statement that two quantities are **not necessarily equal**, built with one of four signs: - $a < b$ – a is **less than** b - $a > b$ – a is **greater than** b - $a \leq b$ – a is **less than or equal to** b (at most) - $a \geq b$ – a is **greater than or equal to** b (at least)

A handy memory aid: the sign is a hungry mouth that always opens toward the **larger** side, so $7 > 3$ and $3 < 7$ say the same thing. The bar underneath ($\leq$, $\geq$) adds "or equal to", quietly including the boundary value itself.

Solving an inequality

Here is the good news. You solve an inequality with the **same balancing moves** you already use for equations: whatever you do to one side you do to the other, peeling away terms until the variable stands alone. Adding, subtracting, and multiplying or dividing by a **positive** number all keep the inequality true and pointing the same way.

There is exactly **one** new rule, and it is the heart of this chapter:

> When you multiply or divide both sides by a **negative** number, **flip** the inequality sign.

Why? Watch a true statement, $3 < 5$. Multiply both sides by -1: the numbers become -3 and -5. On the number line -3 sits to the **right** of -5, so now $-3 > -5$. The "$<$" had to become "$>$" to stay true. Negating reflects every number across zero, which reverses their order.

inequality solution set compound inequality

Worked examples

Example 1

Solve $x + 7 < 12$ and graph the solution.

SOLUTION

1. Subtract 7 from both sides – no sign change, we only subtracted.
2. $x + 7 \ -7 \ < 12 \ -7$, so $x < 5$. – stays <
3. On a number line, draw an open circle at 5 (because 5 itself is *not* included) and shade everything to the left.

$x < 5$

Example 2

Solve $3x \geq 18$.

SOLUTION

1. Divide both sides by 3. Since 3 is **positive**, the sign stays the same.
2. $\frac{3x}{3} \geq \frac{18}{3}$, so $x \geq 6$. ÷3, no flip
3. Graph: a closed (filled) circle at 6 – the bar means 6 is included – shaded to the right.

$x \geq 6$

Example 3

Solve $-2x < 10$.

SOLUTION

1. Divide both sides by -2 to isolate x.
2. Dividing by a **negative** number, so flip $<$ to $>$. flip sign!
3. $x > \frac{10}{-2}$, giving $x > -5$.

$x > -5$

Example 4

Solve $5 - 3x \leq 20$.

SOLUTION

1. Subtract 5 from both sides: $-3x \leq 15$ (still just subtracting, no flip yet).
2. Now divide both sides by -3. Negative divisor, so **flip** $\leq$ to $\geq$. ÷ neg → flip
3. $x \geq \frac{15}{-3}$, so $x \geq -5$.

$x \geq -5$

Example 5

Solve $4x + 1 > x + 13$ (variable on both sides).

SOLUTION

1. Subtract x from both sides: $3x + 1 > 13$. −x both sides
2. Subtract 1: $3x > 12$.
3. Divide by the **positive** 3 – no flip: $x > 4$.

$x > 4$

Example 6

Solve the compound inequality $-1 \leq 2x - 3 < 7$.

SOLUTION

1. A compound inequality is solved by doing the **same step to all three parts** at once. Add 3 everywhere: $-1 + 3 \leq 2x < 7 + 3$, i.e. $2 \leq 2x < 10$. +3 to all 3
2. Divide every part by the positive 2: $1 \leq x < 5$.
3. Graph: closed circle at 1, open circle at 5, shade the segment between them.

$1 \leq x < 5$ – **every number from 1 (included) up to but not including 5.**



Example 6: the solution 1 ≤ x < 5. Closed (filled) circle at 1 includes it; open (hollow) circle at 5 excludes it.

Example 7

Solve $2(1-x) \geq 8$ and graph the solution.

SOLUTION

1. Expand the bracket first: $2 \times 1 - 2 \times x = 2 - 2x$, so the inequality is $2 - 2x \geq 8$. *distribute the 2*
2. Subtract 2 from both sides (only subtracting, no flip yet): $-2x \geq 6$.
3. Divide both sides by -2 – a negative divisor, so **flip** $\geq$ to $\leq$: $x \leq \frac{6}{-2} = -3$. *÷ neg → flip*
4. Graph: a closed circle at -3, shaded to the left.

$x \leq -3$



Example 7: x ≤ −3. Filled circle at −3 (the "or equal to" includes it), shaded leftwards toward −∞.

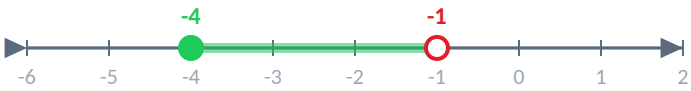
Example 8

Solve the compound inequality $4 < 1 - 3x \leq 13$ (note the negative middle coefficient).

SOLUTION

1. Peel the constant from all three parts: subtract 1 everywhere, giving $4 - 1 < -3x \leq 13 - 1$, i.e. $3 < -3x \leq 12$. *−1 to all 3*
2. Now divide every part by -3. Dividing by a **negative** flips *both* signs **and reverses the order**: $\frac{3}{-3} > x \geq \frac{12}{-3}$, i.e. $-1 > x \geq -4$. *÷ neg → flip both*
3. Rewrite smallest-first for clarity: $-4 \leq x < -1$.

$-4 \leq x < -1$ **– from -4 (included) up to but not including -1.**



Example 8: −4 ≤ x < −1. The negative divisor flipped both signs, so the closed end lands at −4 and the open end at −1.

Example 9

A delivery van weighs 800 kg empty and carries boxes of 25 kg each. A bridge has a limit of 1500 kg. At most how many boxes b can it carry?

SOLUTION

1. Total weight is the empty van plus the boxes: $800 + 25b$ kg. "At most the limit" means $\leq$: $800 + 25b \leq 1500$. *"at most" = ≤*
2. Subtract 800 from both sides: $25b \leq 700$.
3. Divide both sides by the positive 25 – no flip: $b \leq 28$.
4. Boxes must be a whole number, so the largest allowed is 28. *round down to a whole box*

$b \leq 28$, **so at most** 28 **boxes.**

Note: Always isolate the variable *first*, and only check the sign of the number you finally divide or multiply by. Subtracting a negative term (like moving $-3x$ across) is just normal balancing – the flip rule applies **only** to multiplying or dividing the whole inequality by a negative.

Tip: To test your answer, pick any value from your solution set and substitute it into the **original** inequality. If it makes a true statement, your range is pointing the right way. For $-2x < 10$ giving $x > -5$, try $x = 0$: $-2(0) = 0 < 10$ ✓.

⚡ Mental-Math Shortcut

To check instantly whether a solved inequality flipped the right way, **substitute one easy number** – usually 0 – into the original. If $x = 0$ satisfies the original inequality, then 0 must lie in your final solution set; if it does not, 0 must lie *outside* it. For $-2x < 10$: at $x = 0$ the left side is $0 < 10$ ✓, so 0 should be in the answer – and indeed $0 > -5$ is true. One substitution catches a back-to-front sign every time, no re-solving required.

⚠ Common Mistakes

- Forgetting to flip the sign when dividing or multiplying by a negative: solving $-2x < 10$ as $x < -5$ instead of the correct $x > -5$.
- Flipping the sign when it is **not** needed – e.g. flipping just because a negative number *appears*, even though you only added or subtracted it. The flip is triggered only by multiplying/dividing the whole inequality by a negative.
- Drawing the wrong circle: using a filled (closed) circle for $<$ or $>$, or an open circle for $\leq$ or $\geq$. Strict signs get an **open** circle; "or equal to" signs get a **closed** one.
- Shading the wrong side of the number line – read the *final* solved form ($x > -5$ shades right of -5; $x < 5$ shades left of 5).
- In a compound inequality, changing only the middle: you must apply each step to **all three** parts simultaneously.

Quiz

28.1.Q1. Solve $x - 4 > 1$.

A. $x > 5$

B. $x > -3$

C. $x < 5$

D. $x > 3$

28.1.Q2. Solve $-3x \leq 12$.

A. $x \leq -4$

B. $x \geq -4$

C. $x \leq 4$

D. $x \geq 4$

28.1.Q3. Solve $2x + 5 < 17$.

28.1.Q4. Solve $7 - x \geq 3$.

28.1.Q5. True or False: when you divide both sides of an inequality by a positive number, you must reverse the sign.

A. True

B. False

28.1.Q6. On a number line, which circle marks the endpoint of $x \geq 2$?

A. An open (hollow) circle

B. A closed (filled) circle

C. No circle at all

D. A square

28.1.Q7. Solve the compound inequality $0 < x + 2 \leq 5$.

28.1.Q8. Is $x = -1$ a solution of $-2x < 10$?

A. Yes

B. No

🐍 From Concept to Code (Python)

Python's comparison operators `< > <= >=` return a Boolean (`True`/`False`), so you can *test* whether a number satisfies an inequality. Even better, Python lets you write a **chained comparison** that reads exactly like a compound inequality from the page.

```
x = 3
print(x > -5)          # does x satisfy -2x < 10 ?
print(-1 <= x < 5)     # a compound inequality, written naturally
print(2 * x + 5 < 17)  # test 2x + 5 < 17
```

```
output
True
True
True
```

```
# Collect every whole number that satisfies 1 <= x < 5
for x in range(0, 7):
    if 1 <= x < 5:
        print(x, "is in the solution set")
```

```
output
1 is in the solution set
2 is in the solution set
3 is in the solution set
4 is in the solution set
```

Try it:

1. Predict, then run: `print(-3 * 2 <= 12)`. Does it match your solved answer $x \geq -4$ at $x = 2$?
2. Write a chained comparison that is `True` only when `t` is between `0` and `100` inclusive, then test it with `t = 100`.

Answer Key – Chapter 28 – Inequalities

28.1.Q1. $x > 5$ – add 4 to both sides; no flip (only added).

28.1.Q2. $x \geq -4$ – divide by -3, so flip $\leq$ to $\geq$.

28.1.Q3. $x < 6$ – subtract 5 to get $2x < 12$, then divide by 2 (positive, no flip).

28.1.Q4. $x \leq 4$ – subtract 7 to get $-x \geq -4$, then divide by -1 and flip: $x \leq 4$.

28.1.Q5. False – the sign reverses only for a **negative** multiplier or divisor.

28.1.Q6. A closed (filled) circle – the "or equal to" includes 2 itself.

28.1.Q7. $-2 < x \leq 3$ – subtract 2 from all three parts.

28.1.Q8. Yes – $-2(-1) = 2$, and $2 < 10$ is true; it agrees with the solution $x > -5$.

Chapter 29 · Functions & the Coordinate Plane

How to plot points on the x–y plane, treat a function as a machine that turns each input into one output, build a table of values for a rule like f(x)=3x+2, and read a straight-line graph.

The Idea Behind It

Why the graph of $f(x) = mx + c$ is perfectly straight, never bent. *but why?*

A line is exactly a path whose **steepness never changes**: between *any* two of its points you climb the same amount for each step across. That steepness is the **slope**, rise over run:

$$\boxed{\text{slope} = \frac{\text{rise}}{\text{run}} = \frac{y_2 - y_1}{x_2 - x_1}}$$

Take **any** two points on $f(x) = mx + c$, at inputs x_1 and x_2. Their outputs are $y_1 = mx_1 + c$ and $y_2 = mx_2 + c$. Work out the slope between them and watch the $+c$ cancel:

$$\frac{y_2 - y_1}{x_2 - x_1} = \frac{(mx_2 + c) - (mx_1 + c)}{x_2 - x_1} = \frac{m(x_2 - x_1)}{x_2 - x_1} = m$$

The answer is just m with **no x left in it**, so every pair of points gives the **same** slope. Constant steepness everywhere is precisely what makes a graph a straight line, and it is why two points are enough to draw it.

same slope m between all points $\Rightarrow$ straight

$$\therefore\ f(x) = mx + c \text{ has constant slope } m \ \Rightarrow\ \text{a straight line}$$

The c only slides the whole line up or down (it sets where $x = 0$ lands); the slope m alone fixes the tilt. And because each input x yields a single output $mx + c$, no vertical line can ever hit the graph twice, the **vertical-line test** for a genuine function.

So far a letter like x has stood for a *single* unknown number we solve for. A **function** uses a letter differently: as an input you are free to *choose*, which the rule then turns into a matching output. Think of a vending machine – press a button (the input) and out comes one specific snack (the output). Functions are how almost everything in programming, science and money is described: distance from time, price from quantity, brightness from voltage.

The coordinate plane

To *see* a function we draw it on the **coordinate plane**: two number lines crossing at right angles. The horizontal line is the **x-axis**, the vertical line is the **y-axis**, and they meet at the **origin** $(0, 0)$.

ORDERED PAIR

A point is named by an **ordered pair** (x, y). The first number is the **x-coordinate** – how far **right** $(+)$ or left $(-)$ of the origin. The second is the **y-coordinate** – how far **up** $(+)$ or down $(-)$. Order matters: $(3, 1)$ and $(1, 3)$ are different points.

A reliable way to remember the order: you walk along the **hall** (the x-axis, horizontal) *before* you climb the **stairs** (the y-axis, vertical) – "along the corridor, then up the stairs". To plot $(3, 2)$: start at the origin, move 3 right, then 2 up, and mark the point.

What a function is

> **FUNCTION**
>
> A **function** is a rule that assigns to **each** input exactly **one** output. We write $f(x)$, read "f of x", for the output the function f produces from the input x. The set of inputs is the **domain**; the set of outputs is the **range**.

The notation does the bookkeeping for us. If $f(x) = 3x + 2$, then $f(4)$ means "substitute 4 for x": $f(4) = 3(4) + 2 = 14$. The (4) is **not** multiplication – it is the input being fed in. The single most important property:

$$\text{For a function, each input gives } \textbf{exactly one} \text{ output – never two.}$$

function coordinate plane ordered pair linear function

From rule to table to graph

To draw a function, build a small **table of values**: choose a few inputs, compute each output, then plot the resulting ordered pairs $(x, f(x))$ and join them. For a **linear function** $f(x) = mx + b$ the points always line up perfectly straight, so just two or three are enough.

Worked examples

Example 1

Plot the points $A(2, 3)$, $B(-3, 1)$ and $C(0, -2)$.

> **SOLUTION**
>
> 1. For $A(2, 3)$: from the origin go 2 right (positive x), then 3 up (positive y). *hall, then stairs*
> 2. For $B(-3, 1)$: go 3 **left** (negative x), then 1 up.
> 3. For $C(0, -2)$: no horizontal move ($x = 0$ keeps you on the y-axis), then 2 **down**.
>
> **Three points: upper-right, upper-left, and on the y-axis below the origin.**

Example 2

For $f(x) = 3x + 2$, find $f(0)$, $f(1)$ and $f(4)$.

> **SOLUTION**
>
> 1. $f(0) = 3(0) + 2 = 0 + 2 = 2$. *plug in 0*
> 2. $f(1) = 3(1) + 2 = 3 + 2 = 5$.
> 3. $f(4) = 3(4) + 2 = 12 + 2 = 14$. *× before +*
>
> $f(0) = 2$, $f(1) = 5$, $f(4) = 14$.

Example 3

Build a table of values for $f(x) = 3x + 2$ using $x = -1, 0, 1, 2$.

SOLUTION

1. Apply the rule to each input in turn: $f(-1) = 3(-1) + 2 = -1$. − × − = +
2. $f(0) = 2$, $f(1) = 5$, $f(2) = 3(2) + 2 = 8$.
3. Write each result beside its input.

The pairs are $(-1, -1)$, $(0, 2)$, $(1, 5)$, $(2, 8)$. <table class="tt"><thead><tr><th><math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mi>x</mi></mrow><annotation encoding="application/x-tex">x</annotation></semantics></math>x</th><th><math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mi>f</mi><mo stretchy="false">(</mo><mi>x</mi><mo stretchy="false">)</mo><mo>=</mo><mn>3</mn><mi>x</mi><mo>+</mo><mn>2</mn></mrow><annotation encoding="application/x-tex">f(x)=3x+2</annotation></semantics></math>f(x)=3x+2</th><th>point <math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mo stretchy="false">(</mo><mi>x</mi><mo separator="true">,</mo><mtext> </mtext><mi>f</mi><mo stretchy="false">(</mo><mi>x</mi><mo stretchy="false">)</mo><mo stretchy="false">)</mo></mrow><annotation encoding="application/x-tex">(x,\,f(x))</annotation></semantics></math>(x,f(x))</th></tr></thead><tbody><tr><td><math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mo>−</mo><mn>1</mn></mrow><annotation encoding="application/x-tex">-1</annotation></semantics></math>−1</td><td><math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mo>−</mo><mn>1</mn></mrow><annotation encoding="application/x-tex">-1</annotation></semantics></math>

```
</span><span class="katex-html" aria-hidden="true"><span class="base"><span class="strut"
style="height:0.7278em;vertical-align:-0.0833em;"></span><span class="mord">−</span><span
class="mord">1</span></span></span></span></td><td><span class="katex"><span class="katex-
mathml"><math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mo
stretchy="false">(</mo><mo>−</mo><mn>1</mn><mo separator="true">,</mo><mo>−</mo>
<mn>1</mn><mo stretchy="false">)</mo></mrow><annotation encoding="application/x-tex">(-1,-1)
</annotation></semantics></math></span><span class="katex-html" aria-hidden="true"><span
class="base"><span class="strut" style="height:1em;vertical-align:-0.25em;"></span><span class="mopen">
(</span><span class="mord">−</span><span class="mord">1</span><span class="mpunct">,</span><span
class="mspace" style="margin-right:0.1667em;"></span><span class="mord">−</span><span
class="mord">1</span><span class="mclose">)</span></span></span></span></td></tr><tr><td><span
class="katex"><span class="katex-mathml"><math xmlns="http://www.w3.org/1998/Math/MathML">
<semantics><mrow><mn>0</mn></mrow><annotation encoding="application/x-tex">0</annotation>
</semantics></math></span><span class="katex-html" aria-hidden="true"><span class="base"><span
class="strut" style="height:0.6444em;"></span><span class="mord">0</span></span></span></span></td>
<td><span class="katex"><span class="katex-mathml"><math
xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mn>2</mn></mrow><annotation
encoding="application/x-tex">2</annotation></semantics></math></span><span class="katex-html" aria-
hidden="true"><span class="base"><span class="strut" style="height:0.6444em;"></span><span
class="mord">2</span></span></span></span></td><td><span class="katex"><span class="katex-
mathml"><math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mo
stretchy="false">(</mo><mn>0</mn><mo separator="true">,</mo><mn>2</mn><mo stretchy="false">)
</mo></mrow><annotation encoding="application/x-tex">(0,2)</annotation></semantics></math></span>
<span class="katex-html" aria-hidden="true"><span class="base"><span class="strut"
style="height:1em;vertical-align:-0.25em;"></span><span class="mopen">(</span><span
class="mord">0</span><span class="mpunct">,</span><span class="mspace" style="margin-
right:0.1667em;"></span><span class="mord">2</span><span class="mclose">)</span></span></span>
</span></td></tr><tr><td><span class="katex"><span class="katex-mathml"><math
xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mn>1</mn></mrow><annotation
encoding="application/x-tex">1</annotation></semantics></math></span><span class="katex-html" aria-
hidden="true"><span class="base"><span class="strut" style="height:0.6444em;"></span><span
class="mord">1</span></span></span></span></td><td><span class="katex"><span class="katex-
mathml"><math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mn>5</mn>
</mrow><annotation encoding="application/x-tex">5</annotation></semantics></math></span><span
class="katex-html" aria-hidden="true"><span class="base"><span class="strut" style="height:0.6444em;">
</span><span class="mord">5</span></span></span></span></td><td><span class="katex"><span
class="katex-mathml"><math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mo
stretchy="false">(</mo><mn>1</mn><mo separator="true">,</mo><mn>5</mn><mo stretchy="false">)
</mo></mrow><annotation encoding="application/x-tex">(1,5)</annotation></semantics></math></span>
<span class="katex-html" aria-hidden="true"><span class="base"><span class="strut"
style="height:1em;vertical-align:-0.25em;"></span><span class="mopen">(</span><span
```

class="mord">1,5)</td></tr><tr><td><math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mn>2</mn></mrow><annotation encoding="application/x-tex">2</annotation></semantics></math>2</td><td><math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mn>8</mn></mrow><annotation encoding="application/x-tex">8</annotation></semantics></math>8</td><td><math xmlns="http://www.w3.org/1998/Math/MathML"><semantics><mrow><mo stretchy="false">(</mo><mn>2</mn><mo separator="true">,</mo><mn>8</mn><mo stretchy="false">)</mo></mrow><annotation encoding="application/x-tex">(2,8)</annotation></semantics></math>(2,8)</td></tr></tbody></table>

Example 4

Plot the pairs from Example 3 and describe the graph.

SOLUTION

1. Plot $(-1,-1)$, $(0,2)$, $(1,5)$, $(2,8)$ on the plane.
2. Notice they fall on a single straight line – that is why $f(x)=3x+2$ is called *linear*. all in a row
3. The line crosses the y-axis at $(0,2)$, the value of $f(0)$.

A straight line rising left-to-right, passing through $(0,2)$.

Example 5

Read the line: it passes through $(0,1)$ and, for every 1 step right, rises 2 up. Find its rule.

SOLUTION

1. Where the line meets the y-axis is the **intercept** b. Here that is $(0,\ 1\ )$, so $b=1$.
2. How steeply it climbs is the **slope** m: it rises 2 for each 1 across, so $m=2$. rise per step
3. A line is $f(x)=\ m\ x+\ b\ $, so substitute $m=2$ and $b=1$.

$f(x)=\ 2x+1\ $.

Example 6

A taxi charges a £3 flat fee plus £2 per mile. Write this as a function of the miles x and find the cost of a 5-mile trip.

SOLUTION

1. The flat fee is the starting value (the intercept): $b = 3$. The per-mile rate is the slope: $m = 2$.
   fee = intercept
2. So $C(x) = 2x + 3$, where x is miles and C is the cost in pounds.
3. For 5 miles: $C(5) = 2(5) + 3 = 10 + 3 = 13$.

$C(x) = 2x + 3$, **and a 5-mile trip costs £** 13**.**

Example 7

For $f(x) = 2x - 5$, find the input x that gives the output $f(x) = 9$.

SOLUTION

1. So far we picked an input and found the output. Here we run it **backwards**: the output is known, the input is unknown. Set the rule equal to 9: $2x - 5 = 9$. *output known, find input*
2. Solve like an ordinary equation. Add 5 to both sides: $2x = 14$. *+5 both sides*
3. Divide by 2: $x = 7$.
4. Check forwards: $f(7) = 2(7) - 5 = 14 - 5 = 9$ ✓.

The input is $x = 7$ **– the value that the rule sends to** 9**.**

Example 8

Two machines in a row: $f(x) = x + 3$ then $g(x) = 2x$. Feed 4 through f, then its result through g.

SOLUTION

1. Run the first machine on the input 4: $f(4) = 4 + 3 = 7$. *first machine*
2. Take that output 7 and feed it into the **second** machine: $g(7) = 2(7) = 14$. *feed result onward*
3. Writing both steps as one chain: $g(f(4)) = g(7) = 14$. The output of f became the input of g.

The pair of machines turns 4 **into** 14 **(add** 3**, then double).**

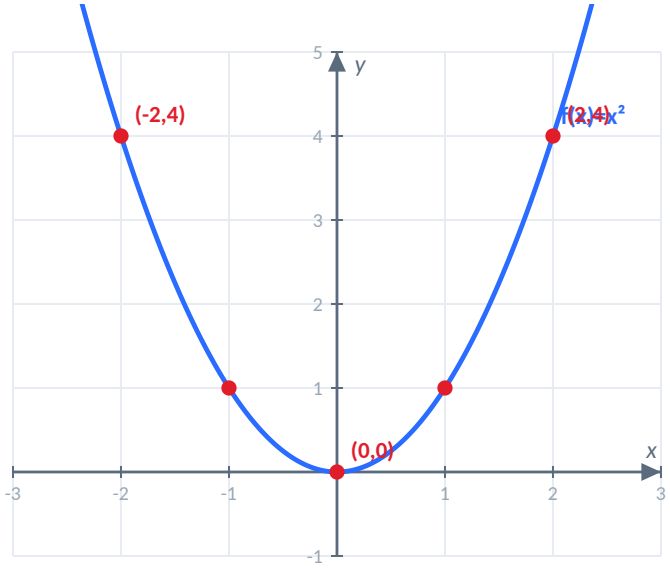
Example 9

The function $f(x) = x^2$ is **not** linear. Build a table for $x = -2, -1, 0, 1, 2$, plot it, and describe the shape.

SOLUTION

1. Square each input – note squaring removes the sign: $f(-2) = (-2)^2 = 4$, $f(-1) = 1$, $f(0) = 0$, $f(1) = 1$, $f(2) = 4$. − × − = +
2. The pairs are $(-2, 4), (-1, 1), (0, 0), (1, 1), (2, 4)$ – symmetric: inputs -2 and 2 share the output 4. ±2 → 4
3. Plotting them, the points do **not** lie on a straight line; they bend into a smooth U-shaped curve (a *parabola*), lowest at the origin.

A symmetric U-shaped curve through $(0, 0)$ – a curved (non-linear) function, yet still exactly one output per input.



Example 9: f(x) = x² plotted from the table. The points fall on a smooth U-shaped parabola, symmetric about the y-axis, with its lowest point at the origin.

Note: Each input has exactly one output, but an output may come from several inputs (a constant function $f(x) = 4$ gives 4 for every input). The "one output per input" test, drawn graphically, is the **vertical line test**: any vertical line may cross the graph of a genuine function at most once.

Mental-Math Shortcut

To evaluate a linear rule $f(x) = mx + b$ in your head, **multiply first, then shift** – exactly the order of operations from Chapter 1. For $f(x) = 3x + 2$ at $x = 7$: see the $\times 3$ first (21), then add the 2 (23). And to jump from one table entry to the **next**, you do not need to recompute: each step of $+1$ in x changes the output by exactly m. So from $f(1) = 5$, the next value $f(2)$ is just $5 + 3 = 8$, then 11, then 14 – counting up by the slope. This "add the slope each time" trick fills a whole table in seconds.

⚠ Common Mistakes

- Reversing the coordinates – plotting (x, y) as y across and x up. Always go along the x-axis (horizontal) **first**, then up/down: $(3, 1) \neq (1, 3)$.
- Reading $f(4)$ as "f times 4". The parentheses hold the **input**; $f(4)$ means evaluate the rule at $x = 4$.
- Mis-ordering operations inside the rule: computing $f(x) = 3x + 2$ at $x = 4$ as $3 \times (4 + 2) = 18$ instead of $3 \times 4 + 2 = 14$. Multiply before adding.
- Sign slips with negative inputs: $f(-1) = 3(-1) + 2 = -1$, not $3(1) + 2 = 5$.
- Confusing slope and intercept: in $f(x) = mx + b$, m is the steepness (rise per step) and b is where the line crosses the y-axis at $x = 0$.

📝 Quiz

29.1.Q1. In the ordered pair $(5, -2)$, what is the y-coordinate?

A. 5

B. -2

C. 3

D. 0

29.1.Q2. For $f(x) = 3x + 2$, what is $f(5)$?

A. 17

B. 25

C. 10

D. 15

29.1.Q3. For $g(x) = 2x - 1$, find $g(0)$.

29.1.Q4. For $f(x) = 3x + 2$, find $f(-2)$.

29.1.Q5. True or False: the point $(0, 4)$ lies on the y-axis.

A. True

B. False

29.1.Q6. In $f(x) = mx + b$, which letter is the slope (the rise per step right)?

A. m

B. b

C. x

D. f

29.1.Q7. A line crosses the y-axis at $(0, 3)$ and rises 4 for every 1 across. Write its rule.

29.1.Q8. True or False: a function may give two different outputs for the same input.

A. True

B. False

🐍 From Concept to Code (Python)

In Python a function is written with `def`: you name it, list its input in parentheses, and `return` the output. The result is the rule $f(x) = 3x + 2$ made real – call it with any input and it hands back exactly one value, just like on paper.

```
def f(x):
    return 3 * x + 2

print(f(0))   # the y-intercept
print(f(1))
print(f(4))
```

```
output
2
5
14
```

```
# Print the table of values from the lesson
for x in range(-1, 3):       # x = -1, 0, 1, 2
    print(x, "->", f(x))
```

```
output
-1 -> -1
0 -> 2
1 -> 5
2 -> 8
```

Try it:

1. Write a function `g(x)` for the taxi cost $C(x) = 2x + 3$, then print the cost of a `5`-mile trip.
2. Loop over `range(0, 5)` and print each ordered pair `(x, f(x))` for `f(x) = 2*x + 1`. Do the outputs go up by the slope each time?

Answer Key – Chapter 29 – Functions & the Coordinate Plane

29.1.Q1. -2 – the **second** number is the y-coordinate (down 2).

29.1.Q2. 17 – $3(5) + 2 = 15 + 2 = 17$.

29.1.Q3. -1 – $2(0) - 1 = -1$; this is also where the line meets the y-axis.

29.1.Q4. -4 – $3(-2) + 2 = -6 + 2 = -4$.

29.1.Q5. True – an x-coordinate of 0 means no horizontal move, so the point sits on the y-axis.

29.1.Q6. m – the slope; b is the y-intercept.

29.1.Q7. $f(x) = 4x + 3$ – slope $m = 4$, intercept $b = 3$.

29.1.Q8. False – by definition each input has exactly **one** output.



PART VII

Algebra in Depth

Algebra grows up: solving several equations at once, working with polynomials, taming exponentials with logarithms, and meeting the structured objects – sequences, matrices, modular clocks – that programming leans on constantly.

Chapter 30 – Systems of Linear Equations · Chapter 31 – Polynomials · Chapter 32 – Exponentials & Logarithms · Chapter 33 – Sequences, Series & Recurrence Relations · Chapter 34 – Modular Arithmetic · Chapter 35 – Sets, Relations & Mappings · Chapter 36 – Matrices & Vectors

Chapter 30 · Systems of Linear Equations

> *How to find the one pair of numbers that satisfies two equations at the same time – by graphing and reading off where the lines cross, and by the algebra methods of substitution and elimination – and how to spot when there is no solution or infinitely many.*

The Idea Behind It

Adding two whole equations together is allowed: it cannot lose the solution. *but why?*

Elimination stacks the two equations and adds them, left to left and right to right, to make a letter vanish. That looks bold, but it rests on a rule you already trust about equals, the **addition property of equality**: if two quantities are equal, you may add equal amounts to both and they stay equal.

$$\boxed{\text{if } P = Q \text{ and } R = S, \text{ then } P + R = Q + S}$$

Take the real system $x + y = 5$ and $x - y = 1$. At the true solution **both** are exact equalities, so the left sides $(x + y)$ and $(x - y)$ equal the right sides 5 and 1. Add the two equations, lining up like with like:

$$\underbrace{(x + y)}_{=5} + \underbrace{(x - y)}_{=1} = 5 + 1$$

On the left the $+y$ and $-y$ cancel, leaving a true statement in one unknown:

$$2x = 6 \;\Rightarrow\; x = 3\,, \qquad \text{then } 3 + y = 5 \;\Rightarrow\; y = 2$$

$$\therefore \text{ the solution } (3,\,2) \text{ survives the addition unchanged.}$$

Because we only ever added equal amounts to equal amounts, every pair that solved the original pair still solves the combined equation, so no solution is created or destroyed. Geometrically the two lines cross at one point, and adding the equations just draws a third line through that **same** crossing.

combine equals with equals

A single equation like $x + y = 5$ has *many* answers: $x = 2, y = 3$ works, and so does $x = 0, y = 5$, and $x = 4, y = 1$. But suppose two things must be true **at once** – say two people's ages add to 5 *and* differ by 1. Now only one pair of numbers fits both demands. Two equations considered together are called a **system**, and solving it means finding the values that make *every* equation true simultaneously.

SYSTEM OF EQUATIONS

A **system of equations** is a set of two or more equations that share the same unknowns. A **solution** is a choice of values for the unknowns that makes **all** the equations true at the same time.

system of equations solution of a system

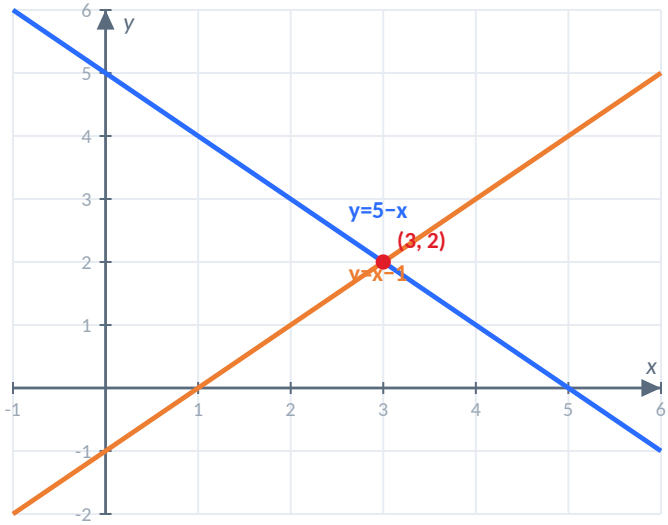
Method 1 – solve by graphing

Each linear equation, redrawn as a line, is the picture of *all* the pairs that satisfy it. So the pair that satisfies **both** equations must lie on **both** lines – it is the point where the two lines **cross**. Graph the two lines, find their

intersection, and read off its coordinates.

Take the system $x + y = 5$ and $x - y = 1$. To graph each, rewrite it as $y = \ldots$:

$$x + y = 5 \;\Rightarrow\; y = 5 - x, \qquad x - y = 1 \;\Rightarrow\; y = x - 1.$$



The blue line is x+y=5; the orange line is x−y=1. They cross at the single point (3, 2) – the solution of the system.

The lines cross at $(3, 2)$, so the solution is $x = 3,\ y = 2$. Check it in **both** original equations: $3 + 2 = 5$ ✓ and $3 - 2 = 1$ ✓. Graphing is wonderfully clear, but it is only as accurate as your drawing, so we also need exact algebra.

Method 2 – substitution

Substitution turns two equations into one. Solve one equation for a single letter, then *substitute* that expression into the other equation. Now only one unknown is left, which you can solve normally; finally back-substitute to get the other.

Method 3 – elimination

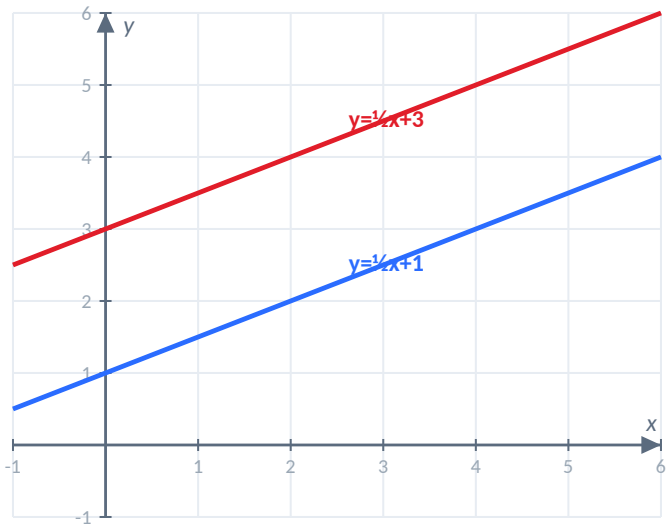
Elimination adds or subtracts the two equations to make one unknown **cancel**. If the coefficients of a letter are equal and opposite (like $+y$ and $-y$), adding the equations removes that letter outright. If they are not, multiply an equation through by a number first so they line up.

> **Note:** Whichever method you use, always finish by checking your pair in **both** original equations. A pair that satisfies only one of them is not a solution of the system.

One, none, or infinitely many

Because two straight lines can meet in only three ways, a linear system has exactly three possibilities:

- The lines **cross once** – exactly **one** solution.
- The lines are **parallel** (same slope, never meet) – **no** solution.
- The lines are **the same line** (one equation is just a multiple of the other) – **infinitely many** solutions.



Two parallel lines (equal slopes, different intercepts) never meet – this system has no solution.

Worked examples

Example 1

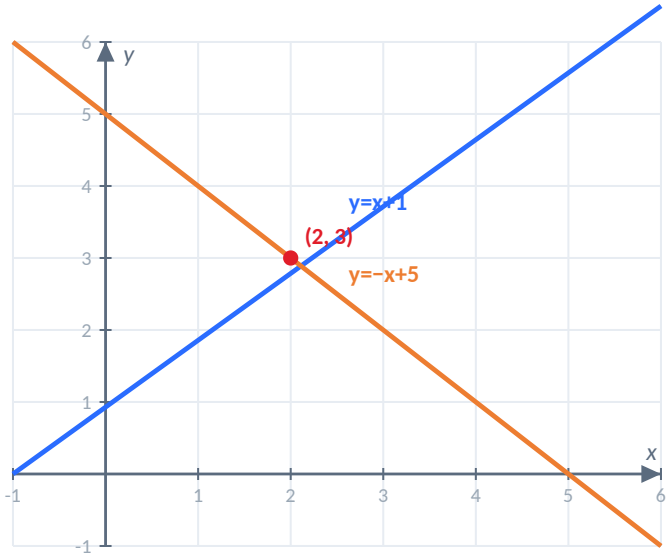
Solve by graphing: $y = x + 1$ and $y = -x + 5$.

SOLUTION

1. Both equations are already in $y = \ldots$ form, so they are easy to plot.
2. The first rises through $(0, 1)$ with slope 1; the second falls through $(0, 5)$ with slope -1.
3. They cross where $x + 1 = -x + 5$. Solving on the picture, the crossing is at $x = 2,\ y = 3$.

   where lines meet
4. Check: $3 = 2 + 1$ ✓ and $3 = -2 + 5$ ✓.

The solution is $(2, 3)$.



Example 1: the lines y=x+1 and y=−x+5 meet at the single point (2, 3).

Example 2

Solve by substitution: $y = 2x$ and $x + y = 9$.

SOLUTION

1. The first equation already gives y on its own: $y = \ 2x$.
2. Substitute $2x$ in place of y in the second equation: $x + \ 2x \ = 9$. *sub in for y*
3. Combine like terms: $3x \ = 9$, so $x = \ 3$.
4. Back-substitute into $y = 2x$: $y = 2(\ 3\ ) = \ 6$.
5. Check: $6 = 2(3)$ ✓ and $3 + 6 = 9$ ✓.

The solution is $x = 3,\ y = 6$.

Example 3

Solve by substitution: $x + y = 7$ and $2x - y = 2$.

SOLUTION

1. Solve the first equation for y: $y = \ 7 - x$.
2. Substitute into the second: $2x - (\ 7 - x\ ) = 2$. *mind the minus*
3. Remove the brackets carefully: $2x \ -7 + x \ = 2$, so $3x - 7 \ = 2$.
4. Add 7 : $3x \ = 9$, so $x = \ 3$.
5. Back-substitute: $y = 7 - \ 3 \ = \ 4$.

The solution is $x = 3,\ y = 4$. **(Check:** $3 + 4 = 7$ ✓, $2(3) - 4 = 2$ ✓.)

Example 4

Solve by elimination: $x + y = 10$ and $x - y = 4$.

SOLUTION

1. The y-terms are $+y$ and $-y$ – equal and opposite, so **add** the two equations. opposite → add
2. Adding the left sides and the right sides: $(x + y) + (x - y) = 10 + 4$, giving $2x = 14$.
3. So $x = 7$.
4. Put $x = 7$ back into $x + y = 10$: $7 + y = 10$, so $y = 3$.

The solution is $x = 7,\ y = 3$. (Check: $7 + 3 = 10$ ✓, $7 - 3 = 4$ ✓.)

Example 5

Solve by elimination (with scaling): $2x + 3y = 12$ and $x - y = 1$.

SOLUTION

1. Make the x-coefficients match. Multiply the second equation by 2: $2x - 2y = 2$. ×2 to line up x
2. Now subtract this from the first to cancel x: $(2x + 3y) - (2x - 2y) = 12 - 2$.
3. The x terms *cancel*: $5y = 10$, so $y = 2$.
4. Back-substitute into $x - y = 1$: $x - 2 = 1$, so $x = 3$.

The solution is $x = 3,\ y = 2$. (Check: $2(3) + 3(2) = 12$ ✓, $3 - 2 = 1$ ✓.)

Example 6

A shop sells apples and bananas. Two apples and one banana cost 4 dollars; one apple and three bananas cost 7 dollars. Find each price.

SOLUTION

1. Let a be the price of an apple and b the price of a banana (in dollars). The two facts become $2a + b = 4$ and $a + 3b = 7$.
2. Solve the first for b: $b = 4 - 2a$.
3. Substitute into the second: $a + 3(4 - 2a) = 7$. sub in for b
4. Expand: $a + 12 - 6a = 7$, so $-5a + 12 = 7$, giving $-5a = -5$ and $a = 1$.
5. Back-substitute: $b = 4 - 2(1) = 2$.

An apple costs 1 dollar and a banana costs 2 dollars. (Check: $2(1) + 2 = 4$ ✓, $1 + 3(2) = 7$ ✓.)

Example 7

Decide how many solutions: $y = 2x + 1$ and $2y = 4x - 6$.

SOLUTION

1. Put the second in $y = \ldots$ form: divide by 2 to get $y = 2x - 3$. ÷2 both sides
2. Compare slopes and intercepts: both have slope 2, but the intercepts ($+1$ and -3) differ.
3. Equal slope with different intercept means the lines are parallel and never meet.

No solution **– the lines are parallel.**

Example 8

Solve by elimination (scaling **both** equations): $3x + 2y = 16$ and $2x + 5y = 18$.

SOLUTION

1. Neither letter lines up yet. Target x: make both coefficients 6 (the lcm of 3 and 2). Multiply the first by 2 and the second by 3.
2. First $\times 2$: $6x + 4y = 32$. Second $\times 3$: $6x + 15y = 54$. match the x terms
3. Subtract the first from the second so x *cancels*: $(6x + 15y) - (6x + 4y) = 54 - 32$, giving $11y = 22$, so $y = 2$.
4. Back-substitute into $3x + 2y = 16$: $3x + 2(2) = 16$, so $3x = 12$ and $x = 4$.

The solution is $x = 4,\ y = 2$**. (Check:** $3(4) + 2(2) = 16$ ✓**,** $2(4) + 5(2) = 18$ ✓**.)**

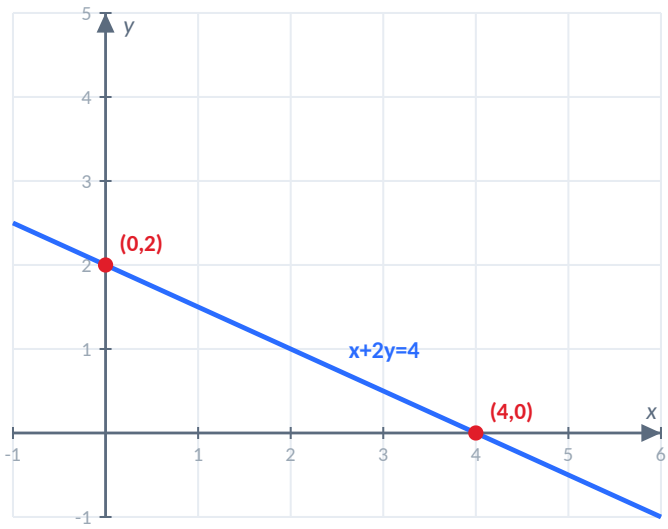
Example 9

Decide how many solutions: $x + 2y = 4$ and $2x + 4y = 8$.

SOLUTION

1. Look for a hidden multiple. Multiply the **first** equation through by 2: $2x + 4y = 8$. ×2 the first
2. This is **identical** to the second equation – the two equations are the *same line* in disguise.
3. Every point on that single line satisfies both, so there is no single crossing point but an endless supply of solutions.

Infinitely many **solutions – one equation is just a multiple of the other.**



Example 9: x+2y=4 and 2x+4y=8 draw the exact same line, so every point on it solves both equations – infinitely many solutions.

Example 10

A cinema sells 200 tickets for 1600 dollars. Adult tickets are 10 dollars and child tickets are 6 dollars. How many of each were sold?

SOLUTION

1. Let a be adult tickets and c child tickets. Total count: $a + c = 200$. Total takings: $10a + 6c = 1600$.
2. Eliminate c. Multiply the count equation by 6: $6a + 6c = 1200$. *scale to match c*
3. Subtract from the money equation so c *cancels*: $(10a + 6c) - (6a + 6c) = 1600 - 1200$, giving $4a = 400$, so $a = 100$.
4. Back-substitute into $a + c = 200$: $100 + c = 200$, so $c = 100$.

100 **adult and** 100 **child tickets. (Check:** $100 + 100 = 200$ ✓, $10(100) + 6(100) = 1600$ ✓.**)**

Mental-Math Shortcut

Pick the method the system hands you. If one equation already has a letter alone on one side (like $y = 2x$ or $x = \ldots$), **substitution** is fastest – just slot it in. If a letter appears as $+n$ in one equation and $-n$ in the other (like $+y$ and $-y$), **elimination by adding** wipes it out in a single step. A quick sanity check before you even solve: rewrite both as $y = mx + c$ and glance at the slopes. **Different slopes** guarantee exactly one crossing; **same slope, different intercept** means no solution; **identical lines** means infinitely many. You can often see the answer's *type* in two seconds.

A system is really a single sentence joined by the word **AND**. Writing the system $x + y = 5$ and $x - y = 1$ asserts the compound claim

$$(x + y = 5)\ \mathbf{AND}\ (x - y = 1).$$

From Chapter 18, an AND is true only when **both** parts are true. So a pair (x, y) solves the system exactly when it satisfies the first equation *and* the second – not one or the other. This is why the graphical answer is the **intersection** of the two lines: the first line is the set of points making the first part true, the second line the set making the second part true, and the points that lie on *both* lines are precisely the points in the **overlap** of those two sets – the logical AND drawn as a picture. It also explains the "no solution" case cleanly: parallel lines share no point, so there is no (x, y) for which both parts are simultaneously true, and the compound AND is false everywhere. Solving a system is therefore the same logical act as asking "for which inputs are *both* conditions satisfied at once?"

⚠ Common Mistakes

- Finding a pair that fits only **one** equation and stopping. A solution must satisfy **both** – always substitute back into each original equation.
- Sign slips when substituting an expression in brackets: $2x - (7 - x)$ is $2x - 7 + x$, **not** $2x - 7 - x$. Distribute the minus sign to every term.
- In elimination, adding when you should subtract (or vice-versa). Add when the matching coefficients are **opposite** ($+y$ and $-y$); subtract when they are the **same** ($+2x$ and $+2x$).
- Forgetting to back-substitute: after finding x, you still owe a value for y (and must report both).
- Calling parallel lines "one solution". Equal slopes with different intercepts give **no** solution; the very same line gives **infinitely many**.

Quiz

30.1.Q1. A solution of a system of two equations must make…

A. the first equation true

B. the second equation true

C. **both** equations true at once

D. either equation true

30.1.Q2. Solve by substitution: $y = 3x$ and $x + y = 8$.

30.1.Q3. Solve by elimination: $x + y = 9$ and $x - y = 3$.

30.1.Q4. On a graph, the solution of a two-line system is…

A. the y-intercept of either line

B. the point where the lines cross

C. the steeper line

D. the origin

30.1.Q5. How many solutions does a system of two **parallel** lines have?

A. exactly one

B. none

C. infinitely many

D. two

30.1.Q6. True or False: if the two equations describe the **same** line, the system has infinitely many solutions.

A. True

B. False

30.1.Q7. Solve: $2x + y = 7$ and $y = x + 1$.

30.1.Q8. You have $x = 5 - 2y$ ready. Which method is most natural to finish the system?

A. graphing

B. substitution

C. elimination

D. guessing

🐍 From Concept to Code (Python)

Two natural jobs for code: **check** whether a proposed pair really solves a system, and **solve** a 2×2 system directly with the elimination formula. Both use nothing but ordinary arithmetic.

```
# Check a candidate solution of:  x + y = 5  AND  x − y = 1
def solves(x, y):
    return (x + y == 5) and (x − y == 1)   # an AND, just like the logic

print(solves(3, 2))   # the real solution
print(solves(4, 1))   # fits the first equation only
```

```
output
True
False
```

```
# Solve  a1 x + b1 y = c1  and  a2 x + b2 y = c2  by elimination
def solve(a1, b1, c1, a2, b2, c2):
    det = a1 * b2 − a2 * b1        # zero means parallel / same line
    if det == 0:
        return None
    x = (c1 * b2 − c2 * b1) / det
    y = (a1 * c2 − a2 * c1) / det
    return (x, y)

print(solve(1, 1, 5, 1, −1, 1))   # x + y = 5,  x − y = 1
```

```
output
(3.0, 2.0)
```

Try it:

1. Use `solves` to confirm $(7, 3)$ solves $x + y = 10$ and $x - y = 4$ but $(8, 2)$ does not.
2. Call `solve` on the apple/banana system ($2a + b = 4$, $a + 3b = 7$) and check it returns $(1.0, 2.0)$. What does `solve` return for two parallel lines, e.g. $x + y = 1$ and $x + y = 2$?

Answer Key – Chapter 30 – Systems of Linear Equations

30.1.Q1. Both equations true at once – that is what "system" means.

30.1.Q2. $x = 2,\ y = 6$ – substitute to get $x + 3x = 8$, so $4x = 8$, then $y = 3(2) = 6$.

30.1.Q3. $x = 6,\ y = 3$ – adding gives $2x = 12$ so $x = 6$, then $y = 9 - 6 = 3$.

30.1.Q4. The point where the lines cross – the only point lying on both lines.

30.1.Q5. None – parallel lines never meet, so no pair satisfies both.

30.1.Q6. True – every point on that shared line satisfies both equations.

30.1.Q7. $x = 2,\ y = 3$ – substitute $x + 1$ for y: $2x + x + 1 = 7$, so $3x = 6$, then $y = 3$.

30.1.Q8. Substitution – one letter is already isolated, so slot $5 - 2y$ into the other equation.

Chapter 31 · Polynomials

How to read, add, subtract and multiply polynomials (including FOIL), recognise the parabola that a quadratic draws, factor simple quadratics, and find a polynomial's roots as the places where its graph crosses the x-axis.

The Idea Behind It

Squaring a sum is not $a^2 + b^2$: the truth is $(a+b)^2 = a^2 + 2ab + b^2$. *but why?*

A square means a bracket times *itself*, $(a+b)(a+b)$, and "times" does not skip the cross-terms. Multiplying every part of one bracket by every part of the other (the **distributive law**) forces a middle term that $a^2 + b^2$ quietly drops:

$$\boxed{(x+y)(c+d) = xc + xd + yc + yd}$$

Set both brackets to $a + b$ and expand all four products (this is exactly **FOIL**):

$$(a+b)^2 = (a+b)(a+b) = a\cdot a + a\cdot b + b\cdot a + b\cdot b$$

The outer and inner products are the **same**, ab and ba, so they join into one doubled term, $ab + ba = 2ab$:

$$= a^2 + 2ab + b^2$$

$$\therefore\ (a+b)^2 = a^2 + 2ab + b^2$$

Picture a square of side $a + b$: it splits into an $a \times a$ tile, a $b \times b$ tile, and **two** $a \times b$ rectangles. Those two rectangles are the $2ab$ that "$a^2 + b^2$" forgets. *the two cross-tiles are real*

You already meet expressions like $3x + 2$ and x^2. Bundle several such powers of x together – added and subtracted – and you have a **polynomial**. Polynomials are the workhorses of algebra: they model the path of a thrown ball, the area of a growing square, the profit of a business. Best of all, the simplest curved ones draw a single, beautiful shape, the **parabola**, which we will learn to read like a map.

POLYNOMIAL

A **polynomial** is a sum of **terms**, where each term is a number (a **coefficient**) times a whole-number power of the variable, such as $4x^3$, $-2x$, or a plain constant 7. Example: $4x^3 - 2x^2 + x - 5$.

Three words name the parts of a term and the whole expression:

- The **coefficient** is the number multiplying the power – in $-2x^2$ it is -2.
- The **degree** is the **highest** power of x that appears – $4x^3 - 2x^2 + x - 5$ has degree 3.
- The **constant term** is the part with no x (an "x^0" term) – here -5.

polynomial coefficient degree root

Degree gives a polynomial its name and its shape. Degree 1 ($ax + b$) is **linear** – a straight line. Degree 2 ($ax^2 + bx + c$) is **quadratic** – a parabola. Degree 3 is **cubic**, and so on.

Adding and subtracting – combine like terms

Two terms are **like terms** when they have the *same* power of x (e.g. $3x^2$ and $5x^2$). To add or subtract polynomials, simply combine like terms and leave the rest. When subtracting, distribute the minus sign to **every** term of the second polynomial first.

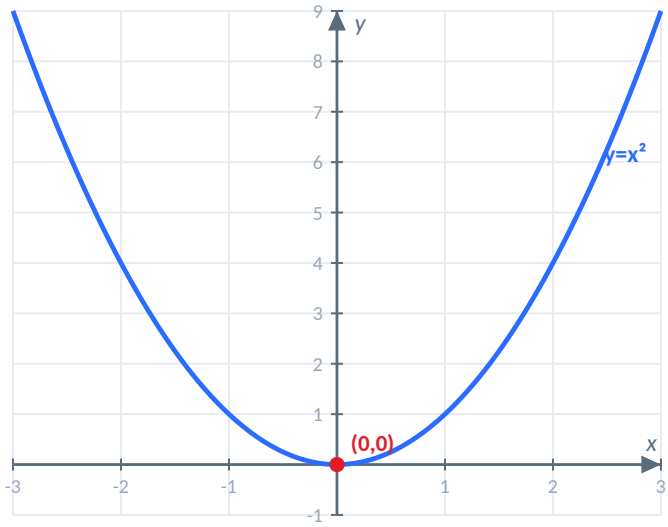
Multiplying – distribute, and FOIL for two binomials

To multiply, every term in the first bracket multiplies every term in the second (the distributive property, stretched). For two two-term brackets (**binomials**) this gives four products, remembered by **FOIL**: **F**irst, **O**uter, **I**nner, **L**ast.

$$(a+b)(c+d) = \underbrace{ac}_{\text{First}} + \underbrace{ad}_{\text{Outer}} + \underbrace{bc}_{\text{Inner}} + \underbrace{bd}_{\text{Last}}$$

The parabola: graphing $y = ax^2 + bx + c$

A quadratic draws a **parabola** – a smooth U-shaped curve. If $a > 0$ the U opens **upward** (a valley); if $a < 0$ it opens **downward** (a hill). The simplest case $y = x^2$ is symmetric about the y-axis, sitting on the origin.

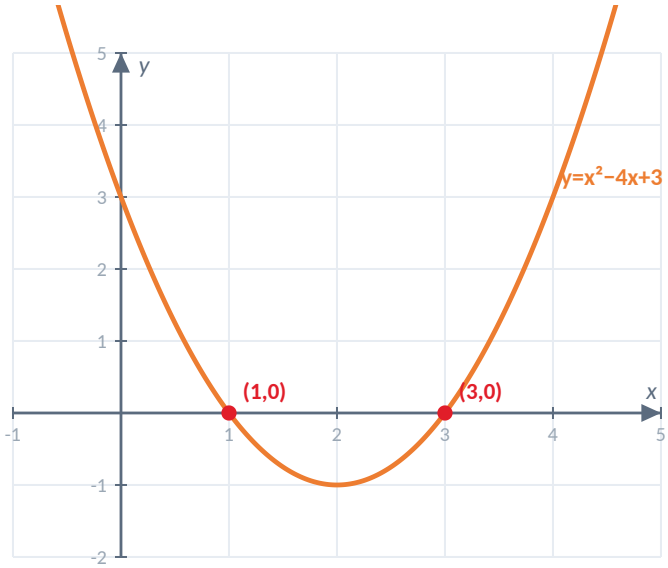


The parabola y = x². It opens upward, is symmetric about the y-axis, and has its lowest point (its vertex) at the origin.

Roots: where the curve crosses the x-axis

A **root** of a polynomial is an x that makes it equal **zero**. On the graph, $y = 0$ means the point sits **on the x-axis**, so the roots are exactly the spots where the curve **crosses** that axis. A quadratic can cross twice, just touch once, or miss entirely.

Factoring reverses multiplication: it rewrites a quadratic as a product of brackets, which instantly reveals the roots. To factor $x^2 + bx + c$, find two numbers that **multiply to** c and **add to** b. Once written as $(x - r_1)(x - r_2)$, the roots are r_1 and r_2, because a product is zero only when one factor is zero (the **zero-product rule**).



y = x² − 4x + 3 factors as (x−1)(x−3). Its roots, 1 and 3, are exactly where the parabola crosses the x-axis.

Note: A handy fact: a polynomial of degree n has **at most** n real roots, so it crosses the x-axis at most n times. A quadratic (degree 2) crosses at most twice – matching the two factors above.

Worked examples

Example 1

State the degree, the leading coefficient, and the constant term of $5x^3 - 2x^2 + 7x - 4$.

SOLUTION

1. Scan for the highest power of x: it is x^3, so the **degree is** 3. *biggest power*
2. The coefficient on that highest power is the **leading coefficient**: 5.
3. The term with no x is the **constant term**: -4.

Degree 3, leading coefficient 5, constant term -4.

Example 2

Add $(3x^2 + 2x - 5) + (x^2 - 4x + 1)$.

SOLUTION

1. Group like terms (same power together). *line up powers*
2. x^2 terms: $3x^2 + x^2 = 4x^2$.
3. x terms: $2x + (-4x) = -2x$.
4. Constants: $-5 + 1 = -4$.

$4x^2 - 2x - 4$.

Example 3

Subtract $(5x^2 - 3x + 2) - (2x^2 + x - 6)$.

SOLUTION

1. Distribute the minus sign across the **whole** second bracket: $-(2x^2 + x - 6) = -2x^2 - x + 6$.

   flip every sign
2. Now add: x^2 terms give $5x^2 - 2x^2 = 3x^2$.
3. x terms give $-3x - x = -4x$.
4. Constants give $2 + 6 = 8$.

$3x^2 - 4x + 8$.

Example 4

Multiply the binomials $(x + 3)(x + 2)$ using FOIL.

SOLUTION

1. **F**irst: $x \cdot x = x^2$. *FOIL*
2. **O**uter: $x \cdot 2 = 2x$.
3. **I**nner: $3 \cdot x = 3x$.
4. **L**ast: $3 \cdot 2 = 6$.
5. Add and combine the like middle terms: $x^2 + 2x + 3x + 6 = x^2 + 5x + 6$.

$x^2 + 5x + 6$.

Example 5

Factor $x^2 + 5x + 6$, then state its roots.

SOLUTION

1. We need two numbers that **multiply to** 6 and **add to** 5. *× to 6, + to 5*
2. Try the factor pairs of 6: 1×6 (sum 7), 2×3 (sum 5) ✓.
3. So $x^2 + 5x + 6 = (x + 2)(x + 3)$ – note this is the reverse of Example 4.
4. Set each factor to zero (zero-product rule): $x + 2 = 0$ gives $x = -2$; $x + 3 = 0$ gives $x = -3$.

$x^2 + 5x + 6 = (x + 2)(x + 3)$, **with roots** $x = -2$ **and** $x = -3$.

Example 6

Factor $x^2 - 4x + 3$ and locate its roots on the graph.

SOLUTION

1. Find two numbers multiplying to $+3$ and adding to -4. Both must be negative: -1 and -3 ($(-1)(-3) = 3, -1 + -3 = -4$) ✓. c>0, b<0 ⇒ both −
2. So $x^2 - 4x + 3 = (x - 1)(x - 3)$.
3. Roots: $x = 1$ and $x = 3$.
4. These are exactly the x-axis crossings drawn in the parabola figure above.

Roots $x = 1$ and $x = 3$ – the points $(1, 0)$ and $(3, 0)$ on the curve.

Example 7

Evaluate the polynomial $p(x) = 2x^2 - 3x + 1$ at $x = 4$.

SOLUTION

1. Substitute 4 for every x: $p(4) = 2(4)^2 - 3(4) + 1$.
2. Powers first (order of operations): $4^2 = 16$, so $2 \cdot 16 = 32$. power before ×
3. Then $-3(4) = -12$.
4. Add the pieces: $32 - 12 + 1 = 21$.

$p(4) = 21$.

Example 8

Square the binomial $(x + 4)^2$ using the pattern.

SOLUTION

1. A square means the bracket times itself: $(x + 4)^2 = (x + 4)(x + 4)$. square = times itself
2. The shortcut pattern is $(x + a)^2 = x^2 + 2ax + a^2$: square the first, **double** the product, square the last.
3. First term x^2; doubled product $2 \cdot 4 \cdot x = 8x$; last term $4^2 = 16$. 2× the product
4. Assemble the three pieces in order.

$(x + 4)^2 = x^2 + 8x + 16$ **– note the middle term $8x$, easy to forget.**

Example 9

Multiply $(x+2)(x^2+3x+1)$ by distributing.

SOLUTION

1. Every term of the first bracket multiplies every term of the second. Start with x across the trinomial: $x \cdot x^2 + x \cdot 3x + x \cdot 1 = x^3 + 3x^2 + x$. *x times each*
2. Now the 2 across the trinomial: $2 \cdot x^2 + 2 \cdot 3x + 2 \cdot 1 = 2x^2 + 6x + 2$. *2 times each*
3. Add the two rows, lining up like powers: $x^3 + (3x^2 + 2x^2) + (x + 6x) + 2$.
4. Combine like terms: $3x^2 + 2x^2 = 5x^2$ and $x + 6x = 7x$.

$x^3 + 5x^2 + 7x + 2$.

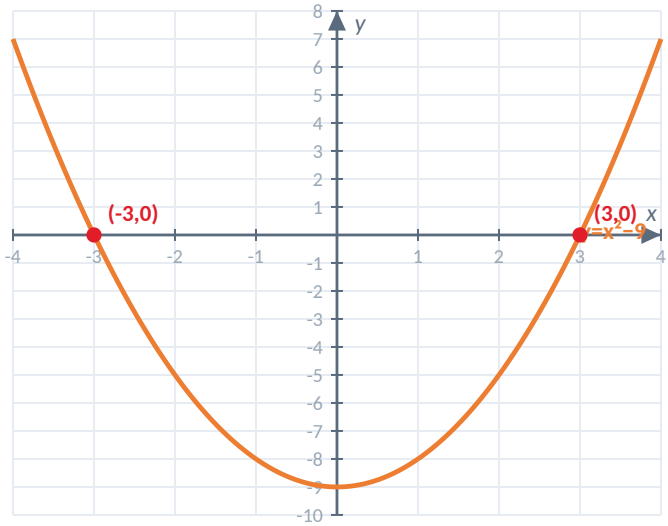
Example 10

Factor the difference of squares $x^2 - 9$ and mark its roots on the graph.

SOLUTION

1. Notice both terms are perfect squares: x^2 and $9 = 3^2$, joined by a **minus**. *spot $a^2 - b^2$*
2. The difference-of-squares pattern is $a^2 - b^2 = (a-b)(a+b)$. With $a = x$ and $b = 3$: $x^2 - 9 = (x-3)(x+3)$.
3. Set each factor to zero: $x - 3 = 0$ gives $x = 3$; $x + 3 = 0$ gives $x = -3$.
4. The two roots sit symmetrically either side of the y-axis – a hallmark of a difference of squares.

$x^2 - 9 = (x-3)(x+3)$, **with roots** $x = 3$ **and** $x = -3$.



Example 10: $y = x^2 - 9 = (x-3)(x+3)$. Its roots -3 and 3 are the two symmetric x-axis crossings, balanced about the y-axis.

⚡ Mental-Math Shortcut

Factoring $x^2 + bx + c$ is a tiny puzzle: find two numbers that multiply to c and add to b. The **signs** point the way instantly. If c is **positive**, both numbers share the **same** sign (both positive if $b > 0$, both negative if $b < 0$). If c is **negative**, the two numbers have **opposite** signs. So for $x^2 - 5x + 6$ you know at a glance to look at two *negative* numbers that multiply to 6 – just -2 and -3. And to **square a binomial** quickly, remember the pattern $(x + a)^2 = x^2 + 2ax + a^2$: middle term is *twice* the product, last term is the square. So $(x + 4)^2 = x^2 + 8x + 16$ in one glance.

Why do the roots of a factored quadratic just *fall out*? It rests on one crisp logical rule about multiplication, the **zero-product rule**:

$$A \cdot B = 0 \quad \Longleftrightarrow \quad A = 0 \ \mathbf{OR} \ B = 0.$$

Read it carefully. A product of two numbers is zero **exactly when at least one factor is zero** – there is no way to multiply two non-zero numbers and land on zero. So once we have written $x^2 - 4x + 3 = (x - 1)(x - 3)$, asking "when is this zero?" becomes "when is $(x - 1) = 0$ **or** $(x - 3) = 0$?" Each piece is a simple equation: $x = 1$ or $x = 3$. The compound condition splits cleanly into two cases joined by **OR** (Chapter 18), and we solve each separately. This is the whole reason factoring is powerful: it converts a hard question about a curved expression into an easy OR of two straight-line questions. The same logic caps the count of roots – two factors can each vanish at one place, so a quadratic has **at most two** roots.

⚠ Common Mistakes

- Combining **unlike** terms: $3x^2 + 2x$ is **not** $5x^2$ or $5x^3$. Only terms with the *same* power may be added.
- Dropping the minus sign in subtraction: $(5x^2 - 3x) - (2x^2 + x)$ is $3x^2 - 4x$, because the $-$ flips **both** terms of the second bracket.
- In FOIL, forgetting the two middle terms: $(x + 3)(x + 2)$ is $x^2 + 5x + 6$, not $x^2 + 6$. The Outer and Inner products must be added.
- Believing $(x + a)^2 = x^2 + a^2$. It is $x^2 + 2ax + a^2$ – the doubled middle term is easy to lose.
- Reading roots with the wrong sign: $(x - 3)$ gives the root $x = +3$ (set the factor to zero), not -3.

Quiz

31.1.Q1. What is the degree of $7x^4 - x^2 + 9$?

A. 4

B. 2

C. 7

D. 9

31.1.Q2. Add: $(2x^2 + 3x - 1) + (x^2 - x + 4)$.

31.1.Q3. Subtract: $(4x^2 + x) - (x^2 - 3x)$.

31.1.Q4. Multiply with FOIL: $(x + 4)(x + 1)$.

A. $x^2 + 5x + 4$

B. $x^2 + 4x + 1$

C. $x^2 + 4$

D. $x^2 + 5x + 5$

31.1.Q5. Factor $x^2 + 7x + 12$.

31.1.Q6. The roots of $(x - 2)(x - 5) = 0$ are…

A. 2 and 5

B. -2 and -5

C. 2 and -5

D. 0 only

31.1.Q7. On a graph, the roots of a polynomial are the points where the curve…

A. crosses the y-axis

B. crosses the x-axis

C. reaches its highest point

D. has slope zero

31.1.Q8. True or False: a quadratic ($\text{degree } 2$) can have at most two real roots.

A. True

B. False

31.1.Q9. Evaluate $p(x) = x^2 - 2x + 3$ at $x = 3$.

🐍 From Concept to Code (Python)

A polynomial is fully described by its **list of coefficients**. Storing $2x^2 - 3x + 1$ as the list `[2, -3, 1]` (highest power first), we can evaluate it for any x with a short loop – the same idea computers use internally.

```
def evaluate(coeffs, x):
    total = 0
    n = len(coeffs) - 1          # highest power
    for i, c in enumerate(coeffs):
        total += c * x ** (n - i)   # c times x^(its power)
    return total

p = [2, -3, 1]                  # means 2x^2 - 3x + 1
print(evaluate(p, 4))           # 2*16 - 3*4 + 1
print(evaluate(p, 0))           # the constant term
```

```
output
21
1
```

```
# Hunt for whole-number roots by testing values
p = [1, -4, 3]                  # x^2 - 4x + 3  =  (x-1)(x-3)
for x in range(-1, 6):
    if evaluate(p, x) == 0:
        print("root at x =", x)
```

```
output
root at x = 1
root at x = 3
```

Try it:

1. Use `evaluate` to compute $x^2 + 5x + 6$ at $x = -2$ and $x = -3$; both should give `0` (they are the roots).
2. Write the polynomial $x^3 - x$ as a list and loop over `range(-3, 4)` to find its three roots.

Answer Key – Chapter 31 – Polynomials

31.1.Q1. 4 – the highest power of x present.

31.1.Q2. $3x^2 + 2x + 3$ – combine like terms: $3x^2$, $3x - x = 2x$, $-1 + 4 = 3$.

31.1.Q3. $3x^2 + 4x$ – the minus flips the second bracket: $4x^2 - x^2 = 3x^2$ and $x - (-3x) = 4x$.

31.1.Q4. $x^2 + 5x + 4$ – F x^2, O x, I $4x$, L 4; the middle is $x + 4x = 5x$.

31.1.Q5. $(x + 3)(x + 4)$ – two numbers multiplying to 12 and adding to 7 are 3 and 4.

31.1.Q6. 2 and 5 – set each factor to zero (zero-product rule).

31.1.Q7. Crosses the x-axis – there $y = 0$, which is what "root" means.

31.1.Q8. True – degree n gives at most n roots, so a quadratic crosses the x-axis at most twice.

31.1.Q9. 6 – $9 - 6 + 3 = 6$ (square first, then the rest).

Chapter 32 · Exponentials & Logarithms

> *How a logarithm answers the question "what power do I need?" – the exact inverse of an exponent – how to switch between the two forms, the three log laws that turn multiplication into addition, and how the graphs of 2^x and $\log_2 x$ mirror each other across the line $y = x$.*

The Idea Behind It

A logarithm turns multiplication into addition: $\log_b(xy) = \log_b x + \log_b y$. *but why?*

A logarithm just *reads off an exponent*. So the magic of $\log(xy) = \log x + \log y$ is really a fact about exponents you already know: when you multiply powers of the same base, the **exponents add**. That single product law carries the whole thing:

$$\boxed{b^m \cdot b^n = b^{m+n}}$$

Name the two logs we are after. Let $m = \log_b x$ and $n = \log_b y$; by the meaning of a logarithm that is the same as saying

$$x = b^{\,m}, \qquad y = b^{\,n}.$$

Now multiply x and y and apply the boxed law, letting the exponents add:

$$xy = b^m \cdot b^n = b^{\,m+n}$$

Read that last line as a logarithm: the power of b that makes xy is $m + n$, i.e. $\log_b(xy) = m + n$. Put back m and n:

$$\therefore\ \log_b(xy) = \log_b x + \log_b y$$

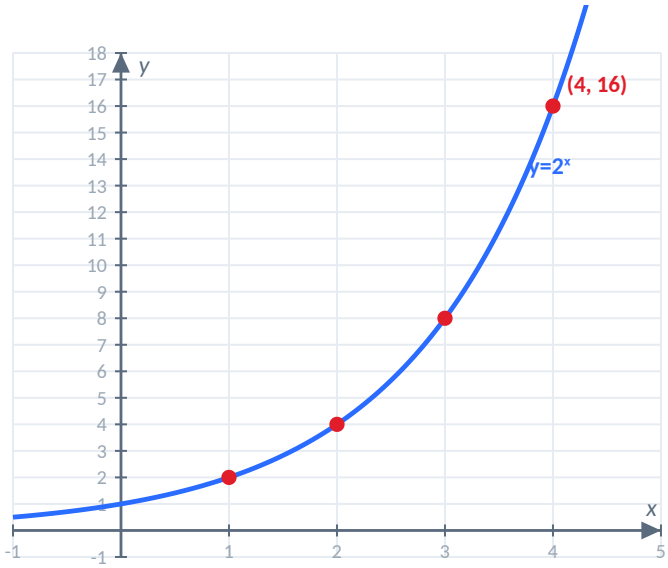
The "$+$" in the answer is nothing but the "add the exponents" rule wearing a logarithm's clothes, which is exactly why log tables and slide rules could replace a hard multiplication with an easy sum.

adding powers becomes adding logs

Repeated multiplication grows astonishingly fast. Fold a paper in half 10 times and its thickness is $2^{10} = 1024$ layers; a single bacterium doubling each hour becomes over a million in a day. This is **exponential growth**, written 2^x, where the variable sits *in the exponent*. Naturally we also want to run the question backwards: "2 to the *what* power gives 1024?" The tool that answers that is the **logarithm**.

Exponential growth, briefly

An **exponential** expression has the variable in the exponent: b^x with a fixed **base** $b > 0$. As x climbs by 1, the value is *multiplied* by b each time. For $b = 2$: $2^1 = 2,\ 2^2 = 4,\ 2^3 = 8,\ 2^4 = 16$ – doubling at every step. The curve starts low, hugs the x-axis on the left, then rockets upward.



Exponential growth $y = 2^x$. Each step right doubles the height: 2, 4, 8, 16. The logarithm runs this picture backwards.

LOGARITHM

The **logarithm** of y to base b, written $\log_b y$, is the **exponent** you must raise b to in order to get y. In one line:

$$\log_b y = x \quad \Longleftrightarrow \quad b^x = y.$$

In words: "$\log_b y$ is the power that turns b into y."

exponential base logarithm inverse

So $\log_2 8 = 3$ because $2^3 = 8$, and $\log_{10} 1000 = 3$ because $10^3 = 1000$. The two forms – **exponential** and **logarithmic** – say exactly the same fact, just rearranged. Reading the box both ways is the single most important skill in this chapter.

Three logarithms you will meet

- $\log_2$ – base 2, the "doubling" log, everywhere in computing.
- $\log_{10}$ – base 10, the **common** log, written plainly as $\log$.
- $\ln$ – base e ($e \approx 2.718$), the **natural** log, the one calculus loves.

The log laws – turning × into +

Because logarithms *are* exponents, they inherit the exponent rules. These three laws let a logarithm break a hard product into an easy sum – the very trick that powered slide rules and log tables for centuries:

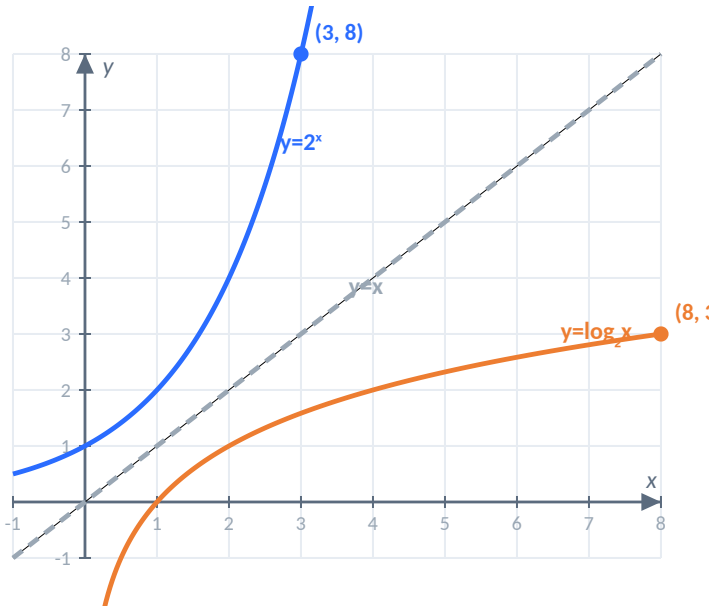
$$\log_b(MN) = \log_b M + \log_b N \quad \text{(a product becomes a sum)}$$

$$\log_b\left(\tfrac{M}{N}\right) = \log_b M - \log_b N \quad \text{(a quotient becomes a difference)}$$

$$\log_b(M^{\,k}) = k\,\log_b M \quad \text{(a power comes down to the front)}$$

Graphs: exponential and log are mirror images

Since the logarithm **undoes** the exponential, their graphs are reflections of each other across the diagonal line $y = x$. The exponential $y = 2^x$ shoots upward; its mirror $y = \log_2 x$ rises slowly and stretches to the right. Whatever input-output pair one curve has, the other has it flipped: $2^3 = 8$ for the exponential becomes $\log_2 8 = 3$ for the log.



$y = 2^x$ (blue) and $y = \log_2 x$ (orange) are mirror images across the dashed line $y = x$. The point (3, 8) on the exponential reflects to (8, 3) on the logarithm.

Note: A logarithm only accepts **positive** inputs. There is no power of 2 that gives 0 or a negative number, so $\log_2 0$ and $\log_2(-4)$ do not exist. On the graph, the orange curve never reaches the left of the y-axis.

Worked examples

Example 1

Find $\log_2 16$.

SOLUTION

1. Ask: "2 to **what** power gives 16?" *log undoes ^*
2. Count the doublings: $2^1 = 2,\ 2^2 = 4,\ 2^3 = 8,\ 2^4 = 16$. *×2 each step*
3. It took 4 doublings, so $\log_2 16 = 4$.

$\log_2 16 = 4$ **(because $2^4 = 16$).**

Example 2

Find $\log_{10} 1000$.

SOLUTION

1. Ask: "10 to what power gives 1000?"
2. $10^{3} = 1000$ (three zeros means the third power). *count the zeros*
3. So the exponent is 3.

$\log_{10} 1000 = 3$.

Example 3

Rewrite the exponential fact $5^2 = 25$ in logarithmic form.

SOLUTION

1. Match the template $b^x = y \Leftrightarrow \log_b y = x$.
2. Here $b = 5$, $x = 2$, $y = 25$.
3. Swap to log form: put y inside and read off the exponent, $\log_5 25 = 2$. *base stays base*

$\log_5 25 = 2$.

Example 4

Solve for x: $\log_3 x = 4$.

SOLUTION

1. Translate to exponential form: $\log_3 x = 4$ means $3^4 = x$. *flip the box*
2. Compute $3^4 = 3 \cdot 3 \cdot 3 \cdot 3 = 81$.

$x = 81$.

Example 5

Use a log law to expand $\log_2(8 \cdot 4)$, then check.

SOLUTION

1. The product law turns log of a product into a sum: $\log_2(8 \cdot 4) = \log_2 8 + \log_2 4$. *× becomes +*
2. Evaluate each: $\log_2 8 = 3$ and $\log_2 4 = 2$.
3. Add: $3 + 2 = 5$.
4. Check directly: $8 \cdot 4 = 32 = 2^5$, so $\log_2 32 = 5$ ✓.

$\log_2(8 \cdot 4) = 5$ **– the product became the sum** $3 + 2$**.**

Example 6

Use the power law to evaluate $\log_{10}(100^3)$.

SOLUTION

1. The power law brings the exponent to the front: $\log_{10}(100^3) = 3 \log_{10} 100$. *power drops down*
2. Now $\log_{10} 100 = 2$ (since $10^2 = 100$).
3. Multiply: $3 \times 2 = 6$.

$\log_{10}(100^3) = 6$. **(Check:** $100^3 = 10^6$**, and** $\log_{10} 10^6 = 6$**.)**

Example 7

A colony of bacteria doubles every hour, starting at 1. After how many hours does it first exceed 1000?

SOLUTION

1. After x hours the count is 2^x. We want the smallest x with $2^x > 1000$.
2. This is a logarithm question: $x > \log_2 1000$. *undo the doubling*
3. Bracket it with known powers: $2^9 = 512$ (too small) and $2^{10} = 1024$ (just over).
4. So $\log_2 1000$ lies between 9 and 10, and the first whole hour past 1000 is 10.

After 10 **hours (**$2^{10} = 1024 > 1000$**).**

Example 8

Use the quotient law to evaluate $\log_2(32 \div 4)$, then check.

SOLUTION

1. The quotient law turns log of a division into a **difference**: $\log_2(32 \div 4) = \log_2 32 - \log_2 4$.
   ÷ becomes −
2. Evaluate each: $\log_2 32 = 5$ (since $2^5 = 32$) and $\log_2 4 = 2$.
3. Subtract: $5 - 2 = 3$.
4. Check directly: $32 \div 4 = 8 = 2^3$, so $\log_2 8 = 3$ ✓.

$\log_2(32 \div 4) = 3$ **– the quotient became the difference** $5 - 2$**.**

Example 9

Combine into a single logarithm: $2\log_3 x + \log_3 5$.

SOLUTION

1. Read the laws **backwards** to build up rather than break down. The coefficient 2 rides up as a power (power law): $2\log_3 x = \log_3 x^2$. *coefficient → power*
2. Now two logs are **added**, so the product law joins them into one log of a product: $\log_3 x^2 + \log_3 5 = \log_3(5x^2)$. *+ becomes ×*

$2\log_3 x + \log_3 5 = \log_3(5x^2)$ **– two laws used in reverse.**

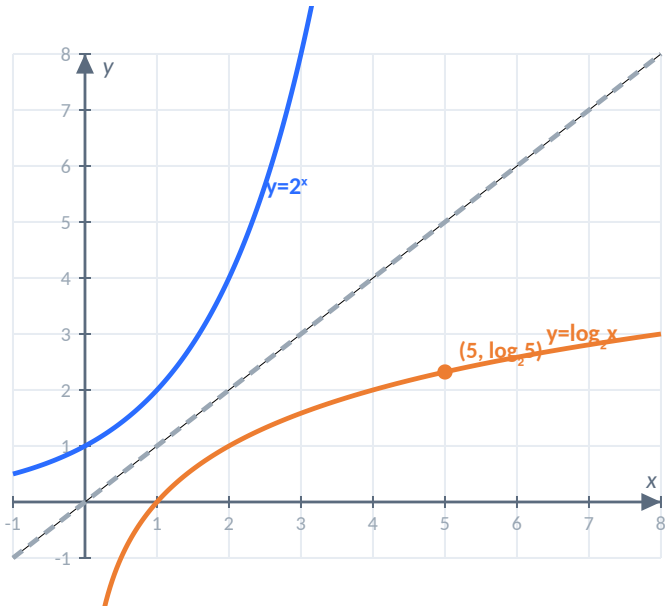
Example 10

Solve $2^x = 5$ to the nearest whole power, then read the answer off the mirror graph.

SOLUTION

1. Take a base-2 logarithm of both sides; on the left the log and the power undo each other: $\log_2(2^x) = x$. *log peels the exponent*
2. So $x = \log_2 5$. Bracket it with known powers: $2^2 = 4$ (too small) and $2^3 = 8$ (too big).
3. Therefore $\log_2 5$ lies between 2 and 3 – closer to 2, since 5 is nearer 4 than 8.
4. On the mirror graph, the point $(5, x)$ on $y = \log_2 x$ reflects the point $(x, 5)$ on $y = 2^x$ – the same fact read off either curve.

$x = \log_2 5 \approx 2.3$, **a value strictly between 2 and 3.**



Example 10: solving $2^x = 5$ means reading $\log_2 5 \approx 2.3$ off the orange curve $y = \log_2 x$ at the input 5 – the mirror image of the exponential.

⚡ Mental-Math Shortcut

A logarithm is just "how many of the base multiply together?" – so count. For powers of 10, the common log is simply the **number of zeros**: $\log_{10} 100 = 2$, $\log_{10} 1{,}000{,}000 = 6$. For base 2, memorise the doubling ladder $1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024$ and the log is the *position* in that ladder: $\log_2 64 = 6$. Two facts work in **every** base and save constant time: $\log_b 1 = 0$ (any base to the power 0 is 1) and $\log_b b = 1$ (the base to the power 1 is itself). And remember the headline trick of the log laws: **a logarithm turns multiplication into addition**, so $\log(ab) = \log a + \log b$ – splitting a big product into a friendly sum.

The whole chapter rests on one logical relationship: exponential and logarithm are **inverse** operations, meaning each *undoes* the other. Symbolically the defining biconditional (an "if and only if", Chapter 20) is

$$\log_b y = x \quad \Longleftrightarrow \quad b^x = y.$$

Read left-to-right and right-to-left: the two statements carry **identical** information, so proving one proves the other. We can watch the undoing happen. Start with a number, raise the base to it, then take the log: $\log_2(2^5)$. By the definition, $\log_2(2^5)$ asks "what power of 2 gives 2^5?" – and the answer is plainly 5, the exponent we started with. So $\log_b(b^x) = x$: the log peels off exactly the exponent the power put on. Going the other way, $b^{\log_b y} = y$ for the same reason. This is precisely what "inverse" means, and it is *why* a logarithm equation can always be rewritten as an exponential one (Examples 3 and 4): we are simply applying the same true biconditional, choosing whichever side is easier to compute.

⚠ Common Mistakes

- Splitting a log of a **sum**. $\log(a + b)$ is **not** $\log a + \log b$. The product law applies to $\log(a \cdot b)$, not $\log(a + b)$.
- Confusing the answer with the input: $\log_2 8 = 3$, **not** 8. The logarithm returns the *exponent*, not the number itself.
- Reversing the box. $\log_b y = x$ means $b^x = y$ – the **base stays the base**. A common slip is writing $x^b = y$.
- Trying to take the log of zero or a negative number. $\log_2 0$ and $\log_2(-5)$ do not exist – no power of a positive base is 0 or negative.
- Treating $\log$ as multiplication, e.g. "cancelling" it. $\log_b(M^k) = k \log_b M$ (the power comes *down*); it is not $(\log_b M)^k$.

Quiz

32.1.Q1. What is $\log_2 8$?

A. 3

B. 4

C. 16

D. 2

32.1.Q2. Rewrite $\log_b y = x$ in exponential form.

32.1.Q3. Find $\log_{10} 10000$.

A. 3

B. 4

C. 5

D. 10

32.1.Q4. Solve for x: $\log_2 x = 5$.

32.1.Q5. Use a law: $\log_3(9 \cdot 27) = ?$

32.1.Q6. What is $\log_b 1$ for any valid base b?

A. 0

B. 1

C. b

D. undefined

32.1.Q7. True or False: the graphs of $y = 2^x$ and $y = \log_2 x$ are reflections of each other across the line $y = x$.

A. True

B. False

32.1.Q8. Why does $\log_2(-4)$ not exist?

32.1.Q9. Apply the power law: $\log_5(25^2) = ?$

🐍 From Concept to Code (Python)

Python's `math` module computes logarithms directly. `math.log2` is base 2, `math.log10` is base 10, and `math.log(y, b)` is base b. Notice how each log *inverts* the matching exponent.

```
import math

print(math.log2(8))          # 2 to what power is 8?
print(math.log10(1000))      # 10 to what power is 1000?
print(math.log(81, 3))       # 3 to what power is 81?
```

```
output
3.0
3.0
4.0
```

```
import math

# The log law  log(a*b) = log(a) + log(b)  checked numerically
left  = math.log2(8 * 4)
right = math.log2(8) + math.log2(4)
print(round(left, 6), round(right, 6))   # equal -> the law holds
```

```
output
5.0 5.0
```

Try it:

1. Print `math.log10(1)` and `math.log2(2)`. Do you get `0.0` and `1.0`, matching the two universal log facts?
2. Loop to find the smallest whole `x` where `2 ** x > 1000` and confirm it equals `math.ceil(math.log2(1000))`.

Answer Key – Chapter 32 – Exponentials & Logarithms

32.1.Q1. 3 – because $2^3 = 8$.

32.1.Q2. $b^x = y$ – the log says "b to the power x gives y".

32.1.Q3. 4 – $10^4 = 10000$ (four zeros).

32.1.Q4. $x = 32$ – exponential form is $2^5 = x = 32$.

32.1.Q5. 5 – product law: $\log_3 9 + \log_3 27 = 2 + 3 = 5$.

32.1.Q6. 0 – any base to the power 0 is 1.

32.1.Q7. True – the log is the inverse of the exponential, so their graphs mirror over $y = x$.

32.1.Q8. No power of 2 gives a negative number; 2^x is always positive, so the logarithm has no positive answer to return.

32.1.Q9. 4 – bring the exponent down: $2 \log_5 25 = 2 \times 2 = 4$.

Chapter 33 · Sequences, Series & Recurrence Relations

How to recognise arithmetic and geometric sequences and write a formula for the nth term, how to add their terms into a series (including sigma notation), and how a recurrence builds an entire sequence from a starting value plus a rule – as with the Fibonacci numbers.

The Idea Behind It

The whole numbers 1 to n add up to $\frac{n(n+1)}{2}$, with no slog. *but why?*

Adding $1 + 2 + \cdots + n$ one term at a time is tedious. The young Gauss spotted a trick: write the sum **forwards**, write it again **backwards** underneath, and add the two rows column by column. Each column then holds a first-with-last pair, and every such pair has the **same** total.

$$\boxed{1 + n = 2 + (n-1) = 3 + (n-2) = \cdots = n + 1}$$

Call the sum S. Stack the forward and reversed copies and add straight down:

$$S = 1 + 2 + 3 + \cdots + n$$
$$S = n + (n-1) + (n-2) + \cdots + 1$$

Every one of the n columns sums to $n + 1$, and there are n of them, so the two rows together give n copies of $(n + 1)$:

$$2S = n\,(n+1)$$

That doubled the sum, so halve it to land back on S:

$$\therefore\ S = 1 + 2 + \cdots + n = \frac{n(n+1)}{2}$$

Check on $n = 100$: the formula gives $\frac{100 \times 101}{2} = 5050$, Gauss's famous answer, found without adding a hundred numbers. *pair the ends, halve the total*

Numbers love to march in patterns: $2, 4, 6, 8, \ldots$ or $3, 9, 27, 81, \ldots$ A list of numbers in a definite order is a **sequence**, and almost every kind of growth – savings earning interest, a salary rising each year, a population multiplying – is a sequence in disguise. Once we can write a formula for the pattern, we can leap straight to the 100th term without listing the first 99.

SEQUENCE

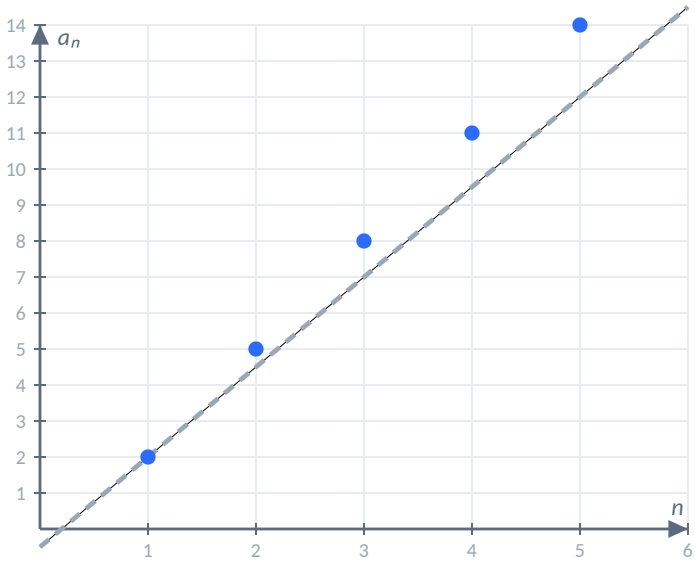
A **sequence** is an ordered list of numbers, called its **terms**. We write the terms $a_1, a_2, a_3, \ldots$, where a_n means the n**th term** – the term in position n.

sequence arithmetic sequence geometric sequence series recurrence

Arithmetic sequences – add a fixed step

In an **arithmetic** sequence you get the next term by **adding** the same number every time. That fixed step is the **common difference** d. The sequence $2, 5, 8, 11, \ldots$ has first term $a_1 = 2$ and common difference $d = 3$. Plotted as points, the terms of an arithmetic sequence lie on a perfectly **straight line** – constant steps make constant slope.

$$a_n = a_1 + (n - 1)\, d \qquad (\text{the } n\text{th term})$$

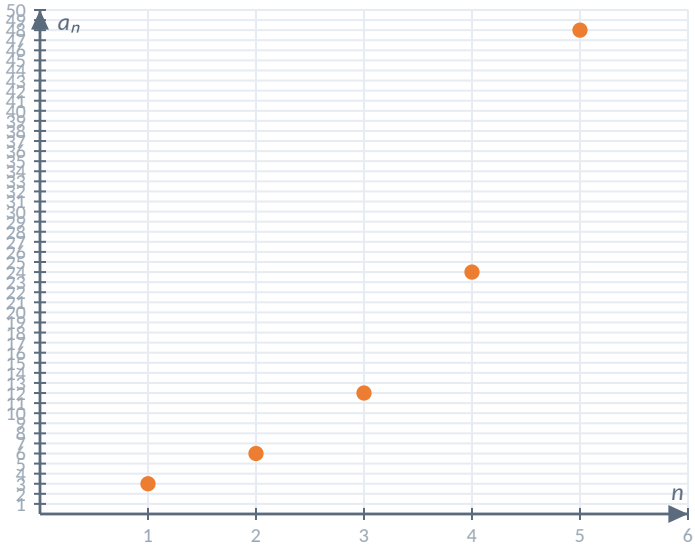


The arithmetic sequence 2, 5, 8, 11, 14 (d = 3). The terms sit on a straight dashed line: equal steps give a constant slope.

Geometric sequences – multiply by a fixed factor

In a **geometric** sequence you get the next term by **multiplying** by the same number every time. That fixed factor is the **common ratio** r. The sequence $3, 6, 12, 24, \ldots$ has $a_1 = 3$ and $r = 2$. Plotted as points, geometric terms **curve upward** ever more steeply – this is exponential growth from Chapter 32, sampled at whole positions.

$$a_n = a_1 \cdot r^{\,n-1} \qquad (\text{the } n\text{th term})$$



The geometric sequence 3, 6, 12, 24, 48 (r = 2). Multiplying by a constant makes the points curve sharply upward.

Series – adding the terms up

A **series** is what you get when you **add** the terms of a sequence. The sum of the first n terms is written with the Greek capital sigma, $\sum$ ("sum of"):

$$\sum_{k=1}^{n} a_k = a_1 + a_2 + a_3 + \cdots + a_n.$$

The notation reads: "add a_k as k runs from 1 up to n." For an arithmetic series there is a famous shortcut – pair the first term with the last, the second with the second-last, and so on; every pair sums to the same total:

$$\sum_{k=1}^{n} a_k = \frac{n}{2}\,(a_1 + a_n) \qquad (\text{arithmetic series sum}).$$

Recurrence relations – a rule plus a start

A **recurrence** defines each term using **earlier** terms, together with a given **starting value**. The arithmetic rule "$a_n = a_{n-1} + 3$ with $a_1 = 2$" is a recurrence. The most celebrated is the **Fibonacci** sequence, where each term is the **sum of the two before it**:

$$F_1 = 1, \quad F_2 = 1, \quad F_n = F_{n-1} + F_{n-2}.$$

This grows $1, 1, 2, 3, 5, 8, 13, 21, \ldots$ – each number the sum of its two predecessors. A short rule plus two seeds generates an endless list.

Note: An *explicit* formula like $a_n = 2 + 3(n-1)$ jumps straight to any term. A *recurrence* like $a_n = a_{n-1} + 3$ needs the previous term first. Both describe the same sequence – the explicit form is faster, the recurrence is often easier to spot.

Worked examples

Example 1

Find the common difference and the next two terms of $7, 10, 13, 16, \ldots$

SOLUTION

1. Subtract consecutive terms to find the step: $10 - 7 = 3$, $13 - 10 = 3$. *same gap each time*
2. The common difference is constant, $d = 3$, so the sequence is arithmetic.
3. Add 3 to keep going: $16 + 3 = 19$, then $19 + 3 = 22$.

$d = 3$; **the next two terms are** 19 **and** 22**.**

Example 2

For the arithmetic sequence $2, 5, 8, 11, \ldots$, find the 10th term.

SOLUTION

1. Identify $a_1 = 2$ and $d = 3$.
2. Use $a_n = a_1 + (n-1)d$ with $n = 10$: $a_{10} = 2 + (10 - 1) \cdot 3$.
3. Work the brackets first: $(10 - 1) \cdot 3 = 9 \cdot 3 = 27$. *9 steps, not 10*
4. Add: $2 + 27 = 29$.

$a_{10} = 29$**.**

Example 3

For the geometric sequence $3, 6, 12, 24, \ldots$, find the 6th term.

SOLUTION

1. Identify $a_1 = 3$ and the ratio $r = 6 \div 3 = 2$. *÷ to find r*
2. Use $a_n = a_1 \cdot r^{n-1}$ with $n = 6$: $a_6 = 3 \cdot 2^{6-1}$.
3. Powers first: $2^5 = 32$.
4. Multiply: $3 \cdot 32 = 96$.

$a_6 = 96$**.**

Example 4

Is $4, 12, 36, 108, \ldots$ arithmetic or geometric? Give the rule.

SOLUTION

1. Check differences: $12 - 4 = 8$ but $36 - 12 = 24$ – not constant, so **not** arithmetic.
2. Check ratios: $12 \div 4 = 3$ and $36 \div 12 = 3$ – constant, so it is **geometric**. ×r each time
3. The common ratio is $r = 3$.

Geometric, with common ratio $r = 3$ (multiply by 3 each time).

Example 5

Find the sum $1 + 2 + 3 + \cdots + 100$.

SOLUTION

1. This is an arithmetic series with $a_1 = 1$, $a_n = 100$, and $n = 100$ terms.
2. Use the pairing shortcut $\sum = \frac{n}{2}(a_1 + a_n)$. pair the ends
3. Substitute: $\frac{100}{2}(1 + 100) = 50 \times 101$.
4. Multiply: $50 \times 101 = 5050$.

The sum is 5050 (the schoolboy Gauss's famous result).

Example 6

Write out the first six Fibonacci numbers from the recurrence $F_1 = 1,\ F_2 = 1,\ F_n = F_{n-1} + F_{n-2}$.

SOLUTION

1. Start with the two seeds: $F_1 = 1, F_2 = 1$.
2. $F_3 = F_2 + F_1 = 1 + 1 = 2$. add the two before
3. $F_4 = F_3 + F_2 = 2 + 1 = 3$.
4. $F_5 = F_4 + F_3 = 3 + 2 = 5$.
5. $F_6 = F_5 + F_4 = 5 + 3 = 8$.

The first six terms are $1, 1, 2, 3, 5, 8$.

Example 7

A salary starts at 30000 dollars and rises by 2000 dollars each year. What is it in year 5, and what is the 5-year total?

SOLUTION

1. Yearly salary is arithmetic: $a_1 = 30000$, $d = 2000$.
2. Year 5 term: $a_5 = 30000 + (5-1) \cdot 2000 = 30000 + 8000 = 38000$ dollars. *4 raises by year 5*
3. Five-year total is the series: $\frac{n}{2}(a_1 + a_5) = \frac{5}{2}(30000 + 38000)$.
4. Compute: $\frac{5}{2} \times 68000 = 2.5 \times 68000 = 170000$.

Year 5 pays 38000 dollars; the five-year total is 170000 dollars.

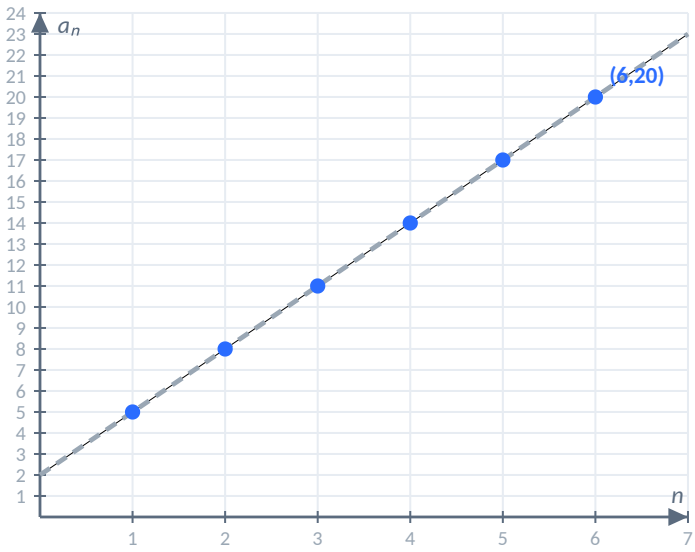
Example 8

Which term of the arithmetic sequence $5, 8, 11, 14, \ldots$ equals 50 ?

SOLUTION

1. Here we know the *value* and want its *position* n – run the nth-term formula backwards. Identify $a_1 = 5$, $d = 3$.
2. Set $a_n = 50$ in $a_n = a_1 + (n-1)d$: $5 + (n-1) \cdot 3 = 50$. *value known, find n*
3. Subtract 5: $(n-1) \cdot 3 = 45$. Divide by 3: $n - 1 = 15$. *−5, then ÷3*
4. Add 1: $n = 16$.

50 is the 16th term. (Check: $a_{16} = 5 + 15 \cdot 3 = 50$ ✓.)



Example 8: the terms 5, 8, 11, … lie on the line $a_n = 3n + 2$. Walking up the dashed line, the value 50 is reached at position n = 16.

Example 9

Sum the first 5 terms of the geometric series $2 + 6 + 18 + 54 + \dots$

SOLUTION

1. Identify $a_1 = 2$ and ratio $r = 6 \div 2 = 3$. ÷ to find r
2. A geometric series has its own shortcut: $\sum_{k=1}^{n} a_1 r^{k-1} = a_1 \dfrac{r^n - 1}{r - 1}$.
3. Substitute $a_1 = 2, r = 3, n = 5$: $2 \cdot \dfrac{3^5 - 1}{3 - 1} = 2 \cdot \dfrac{243 - 1}{2}$. power first
4. Simplify: $2 \cdot \dfrac{242}{2} = 2 \cdot 121 = 242$.

The sum is 242. (Check by adding: $2 + 6 + 18 + 54 + 162 = 242$ ✓.)

Example 10

A recurrence is $a_1 = 1,\ a_n = 2a_{n-1} + 1$. Write out the first five terms.

SOLUTION

1. Start from the seed $a_1 = 1$.
2. $a_2 = 2a_1 + 1 = 2(1) + 1 = 3$. double, then +1
3. $a_3 = 2a_2 + 1 = 2(3) + 1 = 7$.
4. $a_4 = 2a_3 + 1 = 2(7) + 1 = 15$.
5. $a_5 = 2a_4 + 1 = 2(15) + 1 = 31$.

The first five terms are $1, 3, 7, 15, 31$ – each one less than a power of 2 ($2^n - 1$).

⚡ Mental-Math Shortcut

First, name the sequence by what it *does* between terms. Subtract a pair: if the *difference* is the same all along, it is **arithmetic** (think "+ a step"). Divide a pair: if the *ratio* is the same, it is **geometric** (think "× a factor"). For an arithmetic nth term, don't multiply blindly – just count the steps: from a_1 you take $(n - 1)$ steps of d, so a_{10} of $2, 5, 8, \dots$ is $2 + 9 \times 3 = 29$. And to add a run of consecutive whole numbers $1 + 2 + \dots + n$ in your head, use $\frac{n(n+1)}{2}$: for 1 to 100 that is $\frac{100 \times 101}{2} = 5050$ – the average of the ends times how many there are.

A recurrence looks almost too small to pin down an infinite list, yet it does – and the reason is a chain of **if-then** steps, the same engine as mathematical induction (Chapter 22's "for all"). Take Fibonacci: the rule is the conditional "***if*** *I know F_{n-1} and F_{n-2},* ***then*** *I know $F_n = F_{n-1} + F_{n-2}$.*" On its own a rule decides nothing, because it always needs *earlier* terms. So we add the missing ingredient: the **base cases** $F_1 = 1$ and $F_2 = 1$, which are *given*, not derived. Now watch the dominoes fall. We know F_1 and F_2 outright; the rule then *forces* F_3; knowing F_2, F_3 forces F_4; knowing F_3, F_4 forces F_5; and so on without end. Each term is **determined** – there is exactly one value it can be – because every step is a true implication applied to values already nailed down. This is the heart of induction: a

starting truth plus a **truth-preserving step** together guarantee the statement for *every* position n. A finite rule plus a seed therefore defines a genuinely infinite, fully-determined sequence.

⚠ Common Mistakes

- Using n instead of $(n-1)$ in the nth-term formula. From a_1 you take only $(n-1)$ steps, so $a_5 = a_1 + 4d$, not $a_1 + 5d$.
- Mixing up the two types: an **arithmetic** sequence **adds** d; a **geometric** sequence **multiplies** by r. Check a difference *and* a ratio before deciding.
- Confusing a sequence (the list) with a series (the sum). $2, 5, 8$ is a sequence; $2 + 5 + 8 = 15$ is the series.
- Forgetting Fibonacci needs **two** starting values. With only F_1 the rule $F_n = F_{n-1} + F_{n-2}$ cannot begin – it requires F_1 **and** F_2.
- In the arithmetic-sum formula $\frac{n}{2}(a_1 + a_n)$, plugging in the common difference d where the **last term** a_n belongs.

Quiz

33.1.Q1. What is the common difference of $4, 9, 14, 19, \ldots$?

A. 5

B. 4

C. 9

D. 2

33.1.Q2. Find the 8th term of the arithmetic sequence with $a_1 = 3$, $d = 4$.

33.1.Q3. Is $5, 10, 20, 40, \ldots$ arithmetic or geometric, and what is the key number?

33.1.Q4. Find the 5th term of the geometric sequence $2, 6, 18, \ldots$

A. 162

B. 54

C. 108

D. 96

33.1.Q5. What does the symbol $\sum$ tell you to do?

A. multiply the terms

B. add the terms

C. find the difference

D. count the terms

33.1.Q6. Compute $1 + 2 + 3 + \cdots + 50$.

33.1.Q7. In Fibonacci $(1, 1, 2, 3, 5, 8, \ldots)$, what is the term after 8?

A. 13

B. 11

C. 16

D. 10

33.1.Q8. True or False: a recurrence needs both a rule and at least one starting value to define a sequence.

A. True

B. False

33.1.Q9. A sequence is the list of numbers; what is the **series**?

🐍 From Concept to Code (Python)

A loop is the perfect tool for a sequence: each pass produces the next term. Here we generate an arithmetic sequence, then build the Fibonacci numbers straight from their recurrence – the rule becomes a single line inside the loop.

```
# Arithmetic sequence: start 2, common difference 3
a, d = 2, 3
for n in range(1, 6):          # positions 1..5
    print(n, "->", a)
    a = a + d                  # step to the next term
```

```
output
1 -> 2
2 -> 5
3 -> 8
4 -> 11
5 -> 14
```

```
# Fibonacci from the recurrence  F(n) = F(n-1) + F(n-2)
prev, curr = 1, 1              # the two seeds F1, F2
seq = [prev, curr]
for _ in range(6):             # six more terms
    prev, curr = curr, prev + curr   # the rule, in one move
    seq.append(curr)
print(seq)
```

```
output
[1, 1, 2, 3, 5, 8, 13, 21]
```

Try it:

1. Adapt the first loop to a **geometric** sequence: start at 3 and use `a = a * 2` to print $3, 6, 12, 24, 48$.
2. Add up an arithmetic series with a running total: loop `for k in range(1, 101)` accumulating `total += k`, and confirm it prints `5050`.

Answer Key – Chapter 33 – Sequences, Series & Recurrence Relations

33.1.Q1. 5 – each term is 5 more than the previous.

33.1.Q2. 31 – $a_8 = 3 + (8-1) \times 4 = 3 + 28 = 31$.

33.1.Q3. Geometric with common ratio $r = 2$ (each term is doubled).

33.1.Q4. 162 – $a_5 = 2 \cdot 3^4 = 2 \times 81 = 162$ (ratio $r = 3$).

33.1.Q5. Add the terms – $\sum$ means "sum of".

33.1.Q6. 1275 – use $\frac{n(n+1)}{2} = \frac{50 \times 51}{2} = 1275$.

33.1.Q7. 13 – it is the sum of the two before it: $5 + 8 = 13$.

33.1.Q8. True – the rule alone always refers to earlier terms, so a starting value (or values) is required.

33.1.Q9. The sum of those numbers – e.g. the series of $3, 5, 7$ is $3 + 5 + 7 = 15$.

Chapter 34 · Modular Arithmetic

How to do "clock arithmetic" – wrapping numbers around a fixed circle so that, for example, 4 hours after 10 o'clock is 2 – and how this single idea answers questions about even/odd, days of the week, and anything that repeats in a cycle.

The Idea Behind It

You may reduce mod n whenever you like: $(a+b) \bmod n = \big((a \bmod n) + (b \bmod n)\big) \bmod n$.

but why?

Adding two huge numbers and *then* wrapping gives the same answer as wrapping each one *first* and adding the small leftovers. That is what lets a clock add "8 o'clock plus 9 hours" without ever reaching 17. The reason is one fact: a **whole multiple of n is invisible mod n** – it contributes remainder 0.

$$\boxed{(qn) \bmod n = 0 \quad \text{for any whole number } q}$$

Split each number into "whole laps of n" plus a "remainder", writing $a = qn + r$ and $b = q'n + s$, where $r = a \bmod n$ and $s = b \bmod n$. Add them and gather the multiples of n together:

$$a + b = (qn + r) + (q'n + s) = (q + q')\,n + (r + s)$$

The chunk $(q+q')n$ is a whole multiple of n, so it vanishes mod n and only $r + s$ can affect the remainder:

$$(a+b) \bmod n = (r+s) \bmod n$$

$$\therefore\ (a+b) \bmod n = \big((a \bmod n) + (b \bmod n)\big) \bmod n$$

So you are free to shrink the numbers as early as you like, which is exactly why big modular sums stay easy: throw away the laps, keep the leftovers. *the laps don't count*

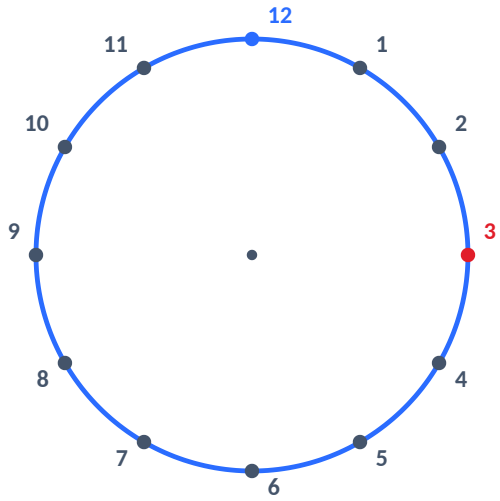
Look at a clock. Three hours after 11 o'clock is not 14 o'clock – it is 2 o'clock. The numbers ran off the end of the 12 and **wrapped around** to the start. This wrapping is one of the most useful ideas in all of computing, and it has a name: **modular arithmetic**. Anything that repeats in a fixed cycle – hours, days of the week, the colours of traffic lights, the position of a wheel – is modular arithmetic in disguise.

The remainder is the whole idea

Take any whole number and divide it by a fixed number m (called the **modulus**). Throw away the quotient and keep only the **remainder**. That remainder is the number "reduced mod m". On a 12-hour clock the modulus is 12; on the days of the week it is 7.

$$15 \div 12 = 1 \text{ remainder } 3, \quad \text{so} \quad 15 \bmod 12 = 3.$$

The remainder is always one of the numbers $0, 1, 2, \ldots, m-1$ – never m itself, and never negative for a positive starting number. So "mod 12" can only ever give you a number from 0 to 11; "mod 7" only 0 to 6. The clock below is exactly the set of possible answers mod 12, bent into a circle.



A clock is the numbers mod 12 wrapped into a circle. Counting 3 steps on from 12 lands on 3, because 15 mod 12 = 3. The top (12) plays the role of 0.

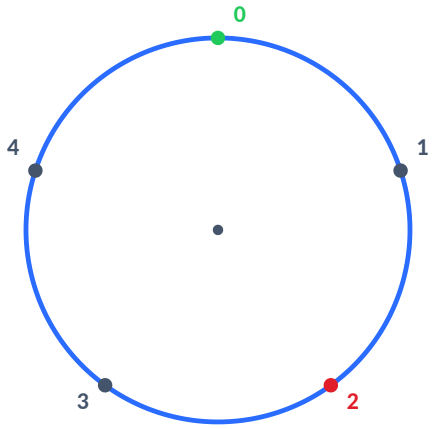
MODULO & MODULUS

For a positive integer m (the **modulus**), $a \bmod m$ is the **remainder** when a is divided by m. It is always one of $0, 1, 2, \ldots, m-1$. Reading: "a mod m". Example: $17 \bmod 5 = 2$, because $17 = 3 \times 5 + 2$.

modulus remainder

Counting on the wheel

To compute $a \bmod m$ by hand, subtract whole copies of m until what remains is below m. You can picture it as a wheel of m slots: walking a steps forward around the wheel from 0 lands you on slot $a \bmod m$. Walk far enough and you come back to where you started – that is the wrap.



A wheel of 5 slots (mod 5). Start at 0 and take 17 steps clockwise: you go round three full times (15 steps) and land 2 further on, at slot 2. So 17 mod 5 = 2.

Congruence – when two numbers land on the same slot

Two numbers that leave the *same* remainder are called **congruent** mod m. On the clock, 3 o'clock and 15 o'clock point the same way, so $15 \equiv 3 \pmod{12}$. The symbol $\equiv$ (three bars) means "is congruent to", and you must always state the modulus in brackets afterwards.

> **CONGRUENCE**
>
> $a \equiv b \pmod{m}$ means a and b leave the **same remainder** mod m – equivalently, that m divides their difference $a - b$ exactly. Example: $15 \equiv 3 \pmod{12}$ because $15 - 3 = 12$, a multiple of 12.

congruent

A friendly everyday case: **even and odd is just arithmetic mod 2.** Every even number leaves remainder 0 when divided by 2; every odd number leaves remainder 1. So "even" means "$\equiv 0 \pmod{2}$" and "odd" means "$\equiv 1 \pmod{2}$". You already do modular arithmetic every time you call a number even.

Unrolled onto a number line, congruence mod 5 means every fifth number carries the *same colour*. The numbers $2, 7, 12$ all reduce to 2; the wheel of five slots is this line rolled up so the colours stack.



Mod 5 on a number line: green dots (0, 5, 10) all reduce to 0; red dots (2, 7, 12) all reduce to 2. Numbers a multiple of 5 apart are congruent – they share a slot on the wheel.

Note: Negative numbers still wrap. To find $-1 \bmod 12$, keep adding 12 until you are in range: $-1 + 12 = 11$, so $-1 \equiv 11 \pmod{12}$. On a clock, "one hour before 12" is 11 – exactly right.

Worked examples

Example 1

Compute $29 \bmod 12$ (think: 29 hours on a clock).

> **SOLUTION**
>
> 1. Subtract whole copies of 12: $29 - 12 = 17$, still too big. *wrap at 12*
> 2. Subtract again: $17 - 12 = 5$, now below 12.
> 3. That used two copies of 12 (24), leaving 5. Check: $29 = 2 \times 12 + 5$.
>
> $29 \bmod 12 = 5$ – **29 hours from 12 o'clock is 5 o'clock.**

Example 2

Compute $45 \bmod 7$.

SOLUTION

1. Find the largest multiple of 7 at or below 45. The 7-times table: $7, 14, 21, 28, 35, 42, 49$.
2. $42 = 6 \times 7$ fits; 49 is too big. *biggest 7× under 45*
3. Subtract: $45 - 42 = 3$.

$45 \bmod 7 = 3$.

Example 3

Is $38 \equiv 14 \pmod{12}$?

SOLUTION

1. Two numbers are congruent mod 12 when their difference is a multiple of 12. *check the gap*
2. Difference: $38 - 14 = 24$.
3. $24 = 2 \times 12$, a multiple of 12 – yes.

True: $38 \equiv 14 \pmod{12}$ **(both leave remainder** 2**).**

Example 4

It is 9 o'clock now. What time is it 100 hours later?

SOLUTION

1. Adding 100 hours then wrapping is the same as adding $(100 \bmod 12)$ hours.
2. $100 \bmod 12$: $96 = 8 \times 12$, so $100 - 96 = 4$. *shrink first*
3. Add 4 hours to 9: $9 + 4 = 13$, which itself wraps: $13 \bmod 12 = 1$.

1 **o'clock. (**$9 + 100 = 109$**, and** $109 \bmod 12 = 1$**.)**

Example 5

Compute $-17 \bmod 5$ (a negative input).

SOLUTION

1. The answer must land in $0, 1, 2, 3, 4$, so add 5 until we get there. *add m until ≥0*
2. $-17 + 5 = -12$, still negative; $+5 = -7$; $+5 = -2$; $+5 = 3$.
3. That added four copies of 5 (20): $-17 + 20 = 3$.

$-17 \bmod 5 = 3$**, i.e.** $-17 \equiv 3 \pmod{5}$**.**

Example 6

Today is Wednesday. What day is it in 30 days?

SOLUTION

1. Days of the week repeat every 7, so work mod 7. Number the days from today: today $= 0$.
2. $30 \bmod 7$: $28 = 4 \times 7$, so $30 - 28 = 2$. *wrap at 7*
3. Count 2 days on from Wednesday: Thursday $(+1)$, Friday $(+2)$.

Friday – **because** $30 \bmod 7 = 2$**, and two days after Wednesday is Friday.**

Example 7

Use mod 2 to decide whether $1{,}234{,}567$ is even or odd.

SOLUTION

1. Even means $\equiv 0 \pmod 2$; odd means $\equiv 1 \pmod 2$.
2. A number's remainder mod 2 depends only on its **last digit**. *last digit only*
3. The last digit is 7, which is odd, so the whole number is odd.

Odd**:** $1{,}234{,}567 \equiv 1 \pmod 2$**.**

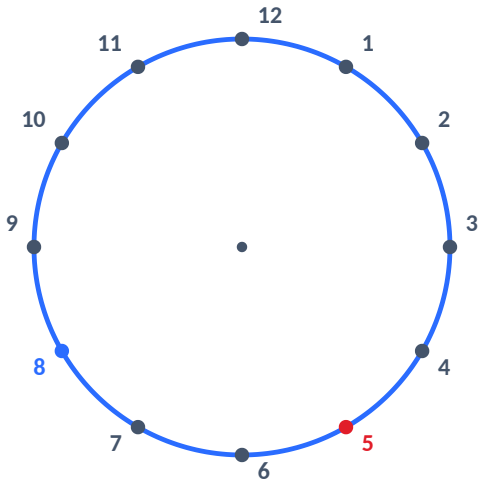
Example 8

Modular **addition**: compute $(8 + 9) \bmod 12$ on the clock.

SOLUTION

1. You may add first, then wrap – or wrap as you go; both agree. Add: $8 + 9 = 17$.
2. Now reduce mod 12: $17 - 12 = 5$, which is below 12. *wrap past 12*
3. On the clock face: start at 8, step 9 hours clockwise, pass the 12, and land on 5.

$(8 + 9) \bmod 12 = 5$ **– 9 hours after 8 o'clock is 5 o'clock.**



Example 8: starting at 8 (blue) and stepping 9 hours clockwise wraps past 12 and lands on 5 (red): (8 + 9) mod 12 = 5.

Example 9

Modular **multiplication**: compute $(7 \times 5) \bmod 12$.

SOLUTION

1. Multiply first as usual: $7 \times 5 = 35$.
2. Reduce mod 12: the largest multiple of 12 at or below 35 is $24 = 2 \times 12$. *biggest 12× under 35*
3. Subtract: $35 - 24 = 11$.

$(7 \times 5) \bmod 12 = 11$ **(because** $35 = 2 \times 12 + 11$**).**

Example 10

A wheel has 7 spokes numbered 0–6. Starting at spoke 4, you turn back (anticlockwise) 20 steps. Which spoke?

SOLUTION

1. Turning back is **subtracting**: the target is $(4 - 20) \bmod 7$.
2. First $4 - 20 = -16$, a negative number, so add copies of 7 until it is in $0 \ldots 6$. *add 7 until ≥0*
3. $-16 + 7 = -9$; $+7 = -2$; $+7 = 5$. That added three copies of 7 (21): $-16 + 21 = 5$.
4. Sanity check forward: from spoke 5, turning **forward** 20 steps gives $(5 + 20) \bmod 7 = 25 \bmod 7 = 4$ ✓.

You land on spoke 5**, i.e.** $(4 - 20) \equiv 5 \pmod 7$**.**

⚡ Mental-Math Shortcut

To reduce a big number mod m, subtract the nearest multiple of m – don't do the long division. For mod 12, lean on the 12-times table ($12, 24, 36, 48, 60, \ldots$): to get $50 \bmod 12$, the nearest lower multiple is 48, so the answer is $50 - 48 = 2$. Two extra free tricks: **mod 10 is just the last digit** ($347 \bmod 10 = 7$), and **mod 2 is even/odd** – look at the last digit only.

Why is it safe to treat 15 o'clock and 3 o'clock as "the same"? Because **congruence mod m is an equivalence relation** – it obeys the same three rules that ordinary equality does, which is exactly what lets us swap one number for a congruent one without breaking anything. Take any whole numbers a, b, c:

- **Reflexive:** $a \equiv a \pmod m$. A number always leaves the same remainder as itself, since $a - a = 0$ is a multiple of m. (Every number is "the same time" as itself.) - **Symmetric:** if $a \equiv b$ then $b \equiv a$. If m divides $a - b$, it also divides $b - a$ (just the negative), so the relationship reads the same both ways. - **Transitive:** if $a \equiv b$ and $b \equiv c$, then $a \equiv c$. Here m divides $a - b$ and m divides $b - c$; add those two differences and the middle term cancels:

$$(a - b) + (b - c) = a - c.$$

A sum of two multiples of m is again a multiple of m, so m divides $a - c$ – meaning a and c share a remainder too.

These three properties (which you met in the language of relations) chop the whole number line into m tidy, non-overlapping groups – the m slots of the wheel. Every number falls into exactly one group, and inside a group all numbers are interchangeable for the purposes of mod-m arithmetic. That is the precise reason "$15 = 3$ on a clock" is not sloppy – it is a statement that they live in the same group.

⚠ Common Mistakes

- Giving the **quotient** instead of the remainder. $17 \bmod 5$ is the leftover 2, **not** the 3 that says how many times 5 fits.
- Producing a result $\geq m$. $14 \bmod 12$ is 2, never 14 – you must wrap until the answer is below the modulus.
- Mishandling negatives by keeping the sign. $-1 \bmod 12$ is 11 (add the modulus), not -1. The answer is always in $0 \ldots m - 1$.
- Writing $\equiv$ without stating the modulus. "$15 \equiv 3$" is meaningless on its own; it is only true *mod* 12 (and false mod 7). Always add $\pmod m$.
- Confusing the symbols: $\equiv$ (congruent, three bars) is not the same as $=$ (equal). $15 \neq 3$, but $15 \equiv 3 \pmod{12}$.

Quiz

34.1.Q1. Compute $20 \bmod 6$.

A. 2

B. 3

C. 4

D. 6

34.1.Q2. Compute $50 \bmod 7$.

34.1.Q3. It is 8 o'clock. What time is it 7 hours later (12-hour clock)?

A. 15

B. 3

C. 1

D. 5

34.1.Q4. What is $-3 \bmod 12$?

34.1.Q5. True or False: $26 \equiv 5 \pmod 7$.

A. True

B. False

34.1.Q6. "Even" means a number is congruent to which value mod 2?

A. 0

B. 1

C. 2

D. either

34.1.Q7. Today is Monday. What day is it in 10 days?

34.1.Q8. What is $347 \bmod 10$?

A. 3

B. 34

C. 7

D. 0

34.1.Q9. True or False: $a \bmod m$ can equal m.

A. True

B. False

🐍 From Concept to Code (Python)

Python writes "mod" with the percent sign `%`. The expression `a % m` gives the remainder of `a` divided by `m` – exactly $a \bmod m$. We can turn that straight into a "what day will it be?" calculator: number the weekdays $0 \ldots 6$, add however many days you like, and let `% 7` do the wrapping.

```
print(17 % 5)        # 17 mod 5  -> remainder 2
print(29 % 12)       # 29 mod 12 -> 5  (29 hours on a clock)
print(-1 % 12)       # Python wraps negatives for you -> 11
print(1234567 % 2)   # last-digit parity: 1 means odd
```

```
output
2
5
11
1
```

```
days = ["Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun"]

def what_day(today_index, days_ahead):
    return days[(today_index + days_ahead) % 7]   # % 7 wraps the week

print(what_day(2, 30))   # start Wed (index 2), 30 days later
print(what_day(0, 365))  # start Mon, one year later
```

```
output
Fri
Tue
```

Try it:

1. Predict, then check: what does `100 % 7` print? (How many days past a whole number of weeks is 100?)
2. Extend `what_day` to also print the time on a 12-hour clock: write `(9 + hours) % 12` for a start of 9 o'clock and try `hours = 100`.

Answer Key – Chapter 34 – Modular Arithmetic

34.1.Q1. 2 – since $18 = 3 \times 6$ and $20 - 18 = 2$.

34.1.Q2. 1 – the nearest lower multiple is $49 = 7 \times 7$, leaving 1.

34.1.Q3. 3 o'clock – $8 + 7 = 15$, and $15 \bmod 12 = 3$.

34.1.Q4. 9 – add the modulus: $-3 + 12 = 9$.

34.1.Q5. True – their difference $26 - 5 = 21 = 3 \times 7$ is a multiple of 7.

34.1.Q6. 0 – even numbers leave remainder 0 mod 2 (odd numbers leave 1).

34.1.Q7. Thursday – $10 \bmod 7 = 3$, and three days after Monday is Thursday.

34.1.Q8. 7 – mod 10 is simply the last digit.

34.1.Q9. False – the remainder is always strictly less than m (between 0 and $m - 1$).

Chapter 35 · Sets, Relations & Mappings

> *How to describe a connection between two collections as a* ***relation*** *(a set of ordered pairs), see the special kind of relation we call a* ***function*** *(each input has exactly one output), and read both as simple arrow diagrams between two columns.*

The Idea Behind It

A set with n elements has exactly 2^n subsets. *but why?*

To *build* a subset, walk through the elements one by one and make a single yes/no decision about each: **in** the subset, or **out**. A subset is nothing more than the record of those choices, so counting subsets means counting choice-sequences. That is the **product rule of counting**: independent choices multiply.

$$\boxed{\text{choices} = c_1 \times c_2 \times \cdots \times c_n}$$

Here every element offers the **same** two options, in or out, and the elements are independent, so we multiply a 2 for each of the n elements:

$$\underbrace{2 \times 2 \times \cdots \times 2}_{n \text{ elements}} = 2^n$$

$$\therefore \text{ number of subsets} = 2^n$$

Check it on $\{a, b\}$ ($n = 2$, so we expect $2^2 = 4$). The four in/out patterns are: out-out $= \varnothing$, in-out $= \{a\}$, out-in $= \{b\}$, in-in $= \{a, b\}$, and indeed

$$\{\varnothing,\ \{a\},\ \{b\},\ \{a, b\}\} \quad (4 \text{ subsets, matching } 2^2)$$

Notice the empty set and the whole set are counted too: choosing "out" for everything, or "in" for everything, are both perfectly good decision-sequences. *each element doubles the count*

You already know a **set** is a collection of distinct things: $A = \{1, 2, 3\}$. But the interesting part of mathematics is usually not the things themselves – it is the **connections between them**. "Is the parent of", "is double of", "is the square root of", "maps the user-id to the username": each of these links items from one set to items from another. The tidy way to record such a link is a **relation**, and the most important relations of all are **functions**. This chapter is the bridge from "what things are there" to "how they are wired together".

A quick set refresher

Recall the core notation, because relations are built directly on top of it:

- $x \in A$ – "x is an **element** of A" (it belongs).
- $A \subseteq B$ – "A is a **subset** of B" – every element of A is also in B.
- $A \cup B$ – the **union**, everything in either set.
- $A \cap B$ – the **intersection**, only what is in both.

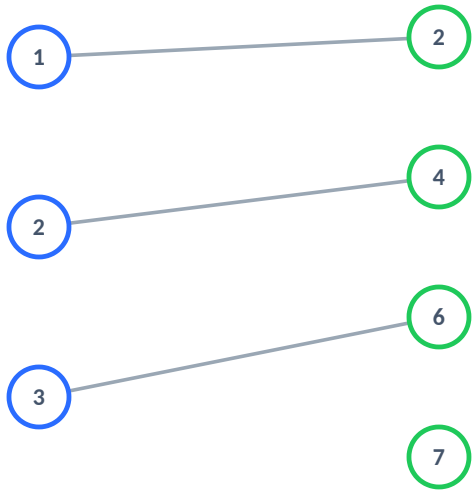
ORDERED PAIR & CARTESIAN PRODUCT

An **ordered pair** (a, b) lists two items *in order*, so $(1, 2) \neq (2, 1)$. The **Cartesian product** $A \times B$ is the set of *all* ordered pairs with the first item from A and the second from B. If $A = \{1, 2\}$ and $B = \{x, y\}$ then $A \times B = \{(1, x), (1, y), (2, x), (2, y)\}$.

ordered pair

A relation is just a set of pairs

A **relation** from A to B picks out *some* of those ordered pairs – the ones where the link holds. Nothing more mysterious than that. If the link is "is double of" between $A = \{1, 2, 3\}$ and $B = \{2, 4, 6, 7\}$, the relation is the set $\{(1, 2), (2, 4), (3, 6)\}$ – it skips 7 because nothing in A doubles to it. We picture a relation as two columns of dots with an **arrow** for every pair.



The relation "is double of" from A={1,2,3} (left) to B={2,4,6,7} (right). Each arrow is one ordered pair: 1→2, 2→4, 3→6. The element 7 receives no arrow.

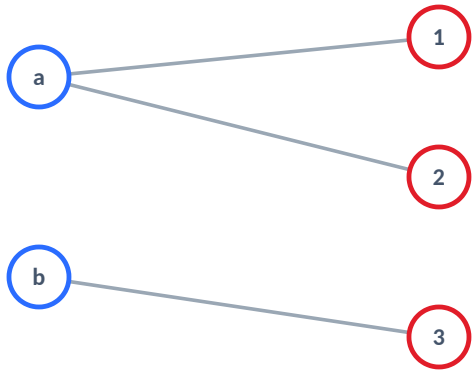
RELATION

A **relation** from set A to set B is any subset of $A \times B$ – that is, any collection of ordered pairs (a, b) with $a \in A$ and $b \in B$. We read (a, b) being in the relation as "a is related to b".

relation

A function is a relation that never forks

A **function** is the special, well-behaved kind of relation where **every input has exactly one output** – every dot on the left has *exactly one* arrow leaving it. No input is left unconnected, and no input fans out to two different outputs. The doubling relation above is a function: each of $1, 2, 3$ sends out a single arrow. The two diagrams below show the difference at a glance.



NOT a function: input a forks to both 1 and 2 (two arrows leave a). A function forbids this – each input may send out only one arrow.

FUNCTION, DOMAIN & RANGE

A **function** is a relation in which each element of the input set is paired with **exactly one** output. The **domain** is the set of all inputs (every left-hand dot); the **range** is the set of outputs actually used (the right-hand dots that receive an arrow). We write $f(a) = b$ to mean "the function f sends input a to output b".

function domain range

A clean test you can apply to any arrow diagram: **a relation is a function exactly when every left-hand dot has one and only one arrow coming out of it.** Two arrows out of the same input → not a function. Zero arrows out of some input → not a function (that input has no output).

Note: It is perfectly fine for two different inputs to share the same output – two arrows *arriving* at one right-hand dot is allowed. The rule only restricts arrows *leaving* each input. So $f(x) = x^2$ is a function even though $f(-2)$ and $f(2)$ both give 4.

Worked examples

Throughout, let $A = \{1, 2, 3\}$ and $B = \{a, b, c\}$.

Example 1

Write out the Cartesian product $A \times B$ for $A = \{1, 2\}$, $B = \{a, b\}$.

SOLUTION

1. Pair every element of A with every element of B, keeping A's element first. *every × every*
2. From 1: $(1, a), (1, b)$. From 2: $(2, a), (2, b)$.

$A \times B = \{(1, a), (1, b), (2, a), (2, b)\}$ – **four pairs (2×2).**

Example 2

The relation R on $\{1, 2, 3, 4\}$ is "is less than". List its pairs.

SOLUTION

1. Include (x, y) whenever $x < y$, both drawn from $\{1, 2, 3, 4\}$. *keep only $x < y$*
2. From 1: $(1, 2), (1, 3), (1, 4)$. From 2: $(2, 3), (2, 4)$. From 3: $(3, 4)$. From 4: nothing is larger.

$R = \{(1, 2), (1, 3), (1, 4), (2, 3), (2, 4), (3, 4)\}$.

Example 3

Is the relation $\{(1, a), (2, b), (3, a)\}$ from A to B a function?

SOLUTION

1. Check each input has exactly one arrow out. Input $1 \to a$ (one), $2 \to b$ (one), $3 \to a$ (one).
   one arrow out each
2. Every input of $A = \{1, 2, 3\}$ appears exactly once as a first item.
3. Two inputs (1 and 3) share the output a – that is allowed.

Yes, **it is a function. Domain $\{1, 2, 3\}$; range $\{a, b\}$ (note c is unused).**

Example 4

Is the relation $\{(1, a), (1, b), (2, c)\}$ a function?

SOLUTION

1. Look for any input that appears twice with different outputs.
2. Input 1 pairs with both a and b – it forks. *two arrows out!*

No **– input 1 has two outputs, breaking the "exactly one" rule.**

Example 5

For the function $f(x) = x^2$ on the domain $\{-2, -1, 0, 1, 2\}$, list the pairs and give the range.

SOLUTION

1. Apply the rule to each input: $f(-2) = 4$, $f(-1) = 1$, $f(0) = 0$, $f(1) = 1$, $f(2) = 4$.
   ±2 both give 4
2. Write each as a pair $(x, f(x))$.
3. The range is the set of distinct outputs.

Pairs: $\{(-2, 4), (-1, 1), (0, 0), (1, 1), (2, 4)\}$; **range** $= \{0, 1, 4\}$. **It is a function even though $-2, 2$ share output 4.**

Example 6

A relation R on $\{1, 2, 3\}$ is $\{(1,1), (2,2), (3,3), (1,2), (2,1)\}$. Is every element related to itself?

SOLUTION

1. "Related to itself" means the pair (x, x) is present for every x in the set. *look for (x, x)*
2. Check: $(1,1)$ yes, $(2,2)$ yes, $(3,3)$ yes.

Yes **– all three self-pairs are present, so R is reflexive (a property explored in The Logic of It).**

Example 7

For $P = \{1, 2, 3, 4\}$ and $Q = \{3, 4, 5\}$, find $P \cup Q$ and $P \cap Q$.

SOLUTION

1. The **union** collects everything in *either* set, listing each element once. Sweep through both: $1, 2, 3, 4$ from P, then add 5 from Q ($3, 4$ already there). *either set, no repeats*
2. So $P \cup Q = \{1, 2, 3, 4, 5\}$.
3. The **intersection** keeps only what is in *both*. The elements common to P and Q are 3 and 4.

only the overlap

4. So $P \cap Q = \{3, 4\}$.

$P \cup Q = \{1, 2, 3, 4, 5\}$ and $P \cap Q = \{3, 4\}$.

Example 8

The function f sends $1 \to b,\ 2 \to a,\ 3 \to c$ from $A = \{1, 2, 3\}$ to $B = \{a, b, c\}$. Draw it as a mapping and confirm it is a function. Give its range.

SOLUTION

1. Picture two columns with one arrow per input. Each of $1, 2, 3$ sends out **exactly one** arrow, so no input forks and none is left bare. *one arrow out each*
2. That is precisely the rule for a function, so f qualifies.
3. The **range** is the set of outputs actually hit. Here every element of B receives an arrow: a, b, c.

f is a function; its range is $\{a, b, c\}$ (all of B is used).



Example 8: the function f: 1→b, 2→a, 3→c. Exactly one arrow leaves each input on the left, so it is a function; every output b, a, c is used, so the range is all of B.

Example 9

Is the relation $R = \{(1,2),(2,1),(2,3)\}$ on $\{1,2,3\}$ symmetric? Is it transitive?

SOLUTION

1. **Symmetric** asks: whenever (a,b) is present, is (b,a) present too? Check each pair: $(1,2)$ has its partner $(2,1)$ ✓; but $(2,3)$ is missing $(3,2)$. need both directions
2. One broken case is enough: R is **not symmetric**.
3. **Transitive** asks: whenever (a,b) and (b,c) are present, is (a,c) present? Chain $(1,2)$ and $(2,3)$ – this demands $(1,3)$, which is $absent$. chain must close
4. So R is **not transitive** either.

R **is** $\text{neither symmetric nor transitive}$ **– each fails on a single missing pair.**

When a relation links a set **to itself**, we can ask three crisp yes/no questions about it. These three properties are the backbone of how mathematics classifies relations – and together they define the "equivalence" relations (like *congruence* from the last chapter, or plain equality).

Let R be a relation on a set S. Read "aRb" as "a is related to b".

- **Reflexive:** *every* element is related to itself – aRa holds for all $a \in S$. Example: "$=$" is reflexive because $a = a$ always. "$<$" is **not** reflexive, since $a < a$ is never true.
- **Symmetric:** the relation reads the same both ways – whenever aRb, also bRa. Example: "is a sibling of" is symmetric. "$<$" is **not**: $2 < 3$ but $3 < 2$ is false.
- **Transitive:** links chain together – whenever aRb and bRc, then aRc. Example: "$<$" *is* transitive, because $a < b$ and $b < c$ force $a < c$.

Now reason through one concrete case to see the machinery work. Take "$=$" on the numbers $\{1, 2, 3\}$. Reflexive? $1{=}1, 2{=}2, 3{=}3$ – yes. Symmetric? If $a{=}b$ then certainly $b{=}a$ – yes. Transitive? Chain two equalities and the middle value passes straight through:

$$a = b \text{ and } b = c \quad \Longrightarrow \quad a = c.$$

Yes again. Because it passes all three tests, equality is an **equivalence relation**: it sorts the set into groups of "the same" things. That is the very same reasoning that justified treating 15 and 3 as interchangeable on a clock – congruence mod 12 passes the identical three-part test.

⚠ Common Mistakes

- Treating (a, b) and (b, a) as the same. Ordered pairs are *ordered*: $(1, 2) \neq (2, 1)$, so "x is double of y" and "y is double of x" are different relations.
- Thinking every relation is a function. A relation is *any* set of pairs; a function is the restricted case where each input has exactly one output.
- Banning shared outputs. Two inputs may map to the **same** output ($f(-2) = f(2) = 4$) and it is still a function – the rule limits arrows *out*, not *in*.
- Confusing **range** with the whole right-hand set. The range is only the outputs actually used; right-hand elements that receive no arrow are excluded.
- Reading $\in$ and $\subseteq$ as the same. A relation is a *subset* of $A \times B$ ($R \subseteq A \times B$); an individual pair is an *element* of it ($(1, 2) \in R$).

Quiz

35.1.Q1. How many pairs are in $A \times B$ if A has 3 elements and B has 4?

A. 7

B. 12

C. 3

D. 4

35.1.Q2. True or False: $(2, 5)$ and $(5, 2)$ are the same ordered pair.

A. True

B. False

35.1.Q3. Is $\{(1, a), (2, a), (3, b)\}$ a function?

A. Yes

B. No

35.1.Q4. Is $\{(1, a), (1, b)\}$ a function?

A. Yes

B. No

35.1.Q5. For $f(x) = x + 1$ on domain $\{0, 1, 2\}$, what is the range?

35.1.Q6. A relation is formally a subset of which set?

A. $A \cup B$

B. $A \cap B$

C. $A \times B$

D. $A \subseteq B$

35.1.Q7. In a function, the set of all *inputs* is called the…

A. range

B. domain

C. union

D. image

35.1.Q8. True or False: in a function, two different inputs may give the same output.

A. True

B. False

35.1.Q9. The relation $\{(1, 1), (2, 2), (3, 3)\}$ on $\{1, 2, 3\}$ has every element related to itself. What is this property called?

A. symmetric

B. reflexive

C. transitive

D. none

🐍 From Concept to Code (Python)

Python mirrors this chapter directly. A mathematical **set** is Python's `set` (membership with `in`, subset with <=). A **function as a mapping** – pairing each input to one output – is a Python `dict` (dictionary): the keys are the domain, the values are the range, and a key can hold only one value, which is exactly the "one output per input" rule.

```
A = {1, 2, 3}
print(2 in A)            # membership  ∈     -> True
print({1, 2} <= A)       # subset      ⊆     -> True

# a relation as a set of ordered pairs (tuples)
double = {(1, 2), (2, 4), (3, 6)}
print((2, 4) in double)  # is 2 related to 4? -> True
```

```
output
True
True
True
```

```
# a function as a dict: each input (key) -> one output (value)
f = {1: "a", 2: "b", 3: "a"}     # the function from Example 3
print(f[3])                       # the output for input 3
print(list(f.keys()))             # the domain
print(set(f.values()))            # the range (distinct outputs)
```

```
output
a
[1, 2, 3]
{'a', 'b'}
```

Try it:

1. A `dict` cannot store two values for one key. Type `g = {1: "a", 1: "b"}; print(g)` and see which output survives – this is Python enforcing the "one output per input" rule.
2. Build the function $f(x) = x^2$ for inputs $-2..2$ as a dict comprehension `{x: x*x for x in range(-2, 3)}`, then print `set(...values())` to read off the range $\{0, 1, 4\}$.

Answer Key – Chapter 35 – Sets, Relations & Mappings

35.1.Q1. 12 – every element of A pairs with every element of B, so 3×4.

35.1.Q2. False – order matters in an ordered pair.

35.1.Q3. Yes – each input $(1, 2, 3)$ has exactly one output; sharing output a is allowed.

35.1.Q4. No – input 1 has two different outputs, which a function forbids.

35.1.Q5. $\{1, 2, 3\}$ – the outputs $f(0) = 1$, $f(1) = 2$, $f(2) = 3$.

35.1.Q6. $A \times B$ – a relation is any set of ordered pairs from the Cartesian product.

35.1.Q7. domain – the inputs; the *range* is the outputs used.

35.1.Q8. True – e.g. x^2 sends both -3 and 3 to 9.

35.1.Q9. reflexive – every (x, x) pair is present.

Chapter 36 · Matrices & Vectors

How to treat a list of numbers as a single object: a **vector** *(a step with direction and length, which you can add and stretch) and a* **matrix** *(a rectangular grid of numbers you can add, scale and multiply) – the data structures behind graphics, games and machine learning.*

The Idea Behind It

Matrix times vector is "each row dotted with the column" – not a coincidence, but the definition unrolled. *but why?*

A matrix is a **recipe of weighted sums**: each entry says *how much of one input feeds one output*. Read a 2×2 matrix that way and the row-times-column rule is forced, no memorising. Write the matrix and the input vector with names:

$$M = \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \qquad \mathbf{v} = \begin{bmatrix} x \\ y \end{bmatrix}$$

The top row $(a\ b)$ is the recipe for the **first output**: "take a of x and b of y." The bottom row $(c\ d)$ is the recipe for the second output. Spelling those weighted sums out:

$$\text{output}_1 = ax + by, \qquad \text{output}_2 = cx + dy$$

But $ax + by$ is exactly the **dot product** of the row $(a\ b)$ with the column $(x\ y)$, and likewise for the second row. Stacking the two outputs:

$$M\mathbf{v} = \begin{bmatrix} ax + by \\ cx + dy \end{bmatrix}$$

$$\therefore\ (\text{row } i) \cdot \mathbf{v} = \text{output } i$$

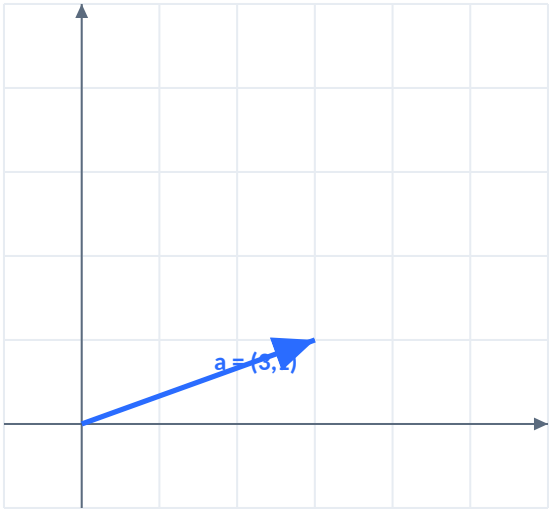
This is also *why* order matters when you multiply two matrices: AB means "do B, then A", and reversing to BA asks the recipes to act in the other order. A quarter-turn then a horizontal stretch leaves a shape differently placed than a stretch then a quarter-turn, so $AB \neq BA$ in general.

a matrix is a list of weighted-sum recipes

So far a number has meant a single quantity. But the moment you describe a position on a screen (x across, y up), a move in a game, or a colour (red, green, blue amounts), you need *several* numbers bundled together and treated as **one thing**. Two such bundles run all of modern computing: the **vector** (a list of numbers in a row or column) and the **matrix** (a whole grid of them). The good news: arithmetic on them is just the ordinary arithmetic you know, done slot by slot.

Vectors: a number with a direction

A **vector** is an arrow: it has a **length** and a **direction**. We pin it down with **components** – how far it goes across (x) and up (y). The vector $\begin{bmatrix} 3 \\ 1 \end{bmatrix}$ means "3 to the right and 1 up". Drawn from the origin, its tip lands at the point $(3, 1)$.



The vector a = (3,1) as an arrow from the origin: 3 across, 1 up. Its length is √(3²+1²) = √10 ≈ 3.16.

VECTOR, COMPONENTS & MAGNITUDE

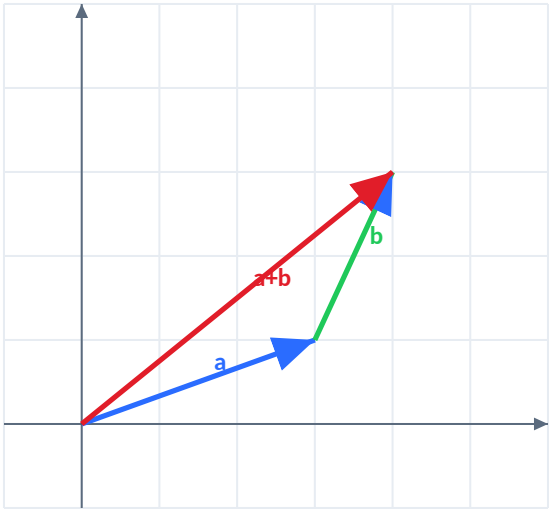
A **vector** is an ordered list of numbers, its **components**, written in a column $\begin{bmatrix} x \\ y \end{bmatrix}$ or a row (x, y). It pictures an arrow with that much horizontal and vertical reach. Its **magnitude** (length) is $|\mathbf{v}| = \sqrt{x^2 + y^2}$, straight from Pythagoras.

vector component magnitude

Adding vectors: tip to tail

To **add** two vectors, add their matching components: the x's together, the y's together. Pictorially you lay the second arrow's tail at the first arrow's tip; the sum is the single arrow from the very start to the very end. Below, $(3, 1) + (1, 2) = (4, 3)$.

$$\begin{bmatrix} 3 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 3 + 1 \\ 1 + 2 \end{bmatrix} = \begin{bmatrix} 4 \\ 3 \end{bmatrix}$$

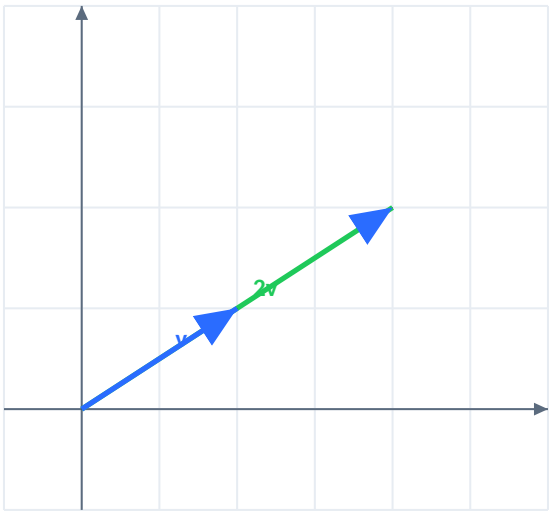


Tip-to-tail addition: place b (green) at the tip of a (blue); the red arrow from origin to the final tip is a+b = (4,3).

Scalar multiplication: stretching an arrow

Multiplying a vector by an ordinary number (called a **scalar** in this setting) stretches or shrinks it, multiplying *each* component. Doubling $(2, 1)$ gives $(4, 2)$ – same direction, twice as long. A negative scalar also flips the arrow to point the opposite way.

$$2 \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}$$



Scalar multiple: 2v (green) points the same way as v (blue) but is twice as long. Each component was doubled.

scalar

Matrices: a grid of numbers

A **matrix** is a rectangular grid of numbers arranged in **rows** and **columns**. Its **size** (or order) is stated as rows $\times$ columns. The matrix below is a 2×2 (two rows, two columns); a vector is really just a matrix with a single column.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$$

A 2×2 matrix: two rows, two columns. The entry in row 2, column 1 is the number 3.

MATRIX

A **matrix** is a rectangular array of numbers in rows and columns; an $m \times n$ matrix has m rows and n columns. The number in row i, column j is called an **entry**. Matrices store grids of data – pixels, tables, the connections in a network – as one object.

matrix

Adding matrices works exactly like vectors: they must be the same size, and you add entry by entry. **Scalar multiplication** multiplies every entry by the number.

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} + \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix} = \begin{bmatrix} 6 & 8 \\ 10 & 12 \end{bmatrix}$$

Multiplying a matrix by a vector

The operation that makes matrices powerful is **matrix multiplication**. Start with the simplest, most useful case: a 2×2 matrix times a column vector. Take each **row** of the matrix, pair it up with the vector, multiply matching numbers and add (a "**row dotted with the column**"). That single number becomes one entry of the answer.

$$\begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 2 \cdot 1 + 0 \cdot 1 \\ 0 \cdot 1 + 3 \cdot 1 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

That matrix stretched the point $(1, 1)$ into $(2, 3)$ – it scaled x by 2 and y by 3. **A matrix is a machine that transforms vectors**: rotations, scalings and reflections in graphics are all "multiply every point by this one matrix". That is why moving a 3-D model on screen is, underneath, matrix arithmetic.

Note: Order matters in matrix multiplication: for a row dotted with a column to make sense, the row and the column must have the same number of entries. A 2×2 matrix multiplies a 2-entry vector, and the result is again a 2-entry vector.

Worked examples

Example 1

Add the vectors $\mathbf{a} = (5, 2)$ and $\mathbf{b} = (-1, 4)$.

SOLUTION

1. Add matching components separately. *slot by slot*
2. x: $5 + (-1) = 4$.
3. y: $2 + 4 = 6$.

$\mathbf{a} + \mathbf{b} = (4, 6)$.

Example 2

Find $3\mathbf{v}$ where $\mathbf{v} = (2, -3)$.

SOLUTION

1. Multiply *each* component by the scalar 3. *hit every part*
2. $3 \times 2 = 6$ and $3 \times (-3) = -9$.

$3\mathbf{v} = (6, -9)$ – **same direction, three times as long.**

Example 3

Find the magnitude (length) of $\mathbf{v} = (3, 4)$.

SOLUTION

1. Use $|\mathbf{v}| = \sqrt{x^2 + y^2}$. *Pythagoras*
2. Square the components: $3^2 = 9$, $4^2 = 16$.
3. Add and take the root: $\sqrt{9 + 16} = \sqrt{25}$.

$|\mathbf{v}| = 5$ – **the classic 3-4-5 right triangle.**

Example 4

Add the matrices $\begin{bmatrix} 2 & 1 \\ 0 & 5 \end{bmatrix}$ and $\begin{bmatrix} 3 & -1 \\ 4 & 2 \end{bmatrix}$.

SOLUTION

1. Same size (2×2), so add entry by entry in matching positions. *match the slots*
2. Top row: $2 + 3 = 5$ and $1 + (-1) = 0$.
3. Bottom row: $0 + 4 = 4$ and $5 + 2 = 7$.

$\begin{bmatrix} 5 & 0 \\ 4 & 7 \end{bmatrix}$.

Example 5

Compute the scalar multiple $2 \begin{bmatrix} 1 & -2 \\ 3 & 0 \end{bmatrix}$.

SOLUTION

1. Multiply every entry by 2. *zeros too*
2. Top row: $2 \times 1 = 2$, $2 \times (-2) = -4$.
3. Bottom row: $2 \times 3 = 6$, $2 \times 0 = 0$.

$\begin{bmatrix} 2 & -4 \\ 6 & 0 \end{bmatrix}$.

Example 6

Multiply the matrix $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ by the vector $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$.

SOLUTION

1. Dot each row of the matrix with the column $(1, 0)$: multiply matching entries, then add. *row · column*
2. Row 1 $(1, 2)$ with $(1, 0)$: $1 \cdot 1 + 2 \cdot 0 = 1$.
3. Row 2 $(3, 4)$ with $(1, 0)$: $3 \cdot 1 + 4 \cdot 0 = 3$.

$\begin{bmatrix} 1 \\ 3 \end{bmatrix}$ – **multiplying by $(1, 0)$ just reads off the first column.**

Example 7

A scaling matrix $\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$ acts on the point $(3, -1)$. Where does it land?

SOLUTION

1. Dot each row with the column $(3, -1)$. *row · column*
2. Row 1 $(2, 0)$: $2 \cdot 3 + 0 \cdot (-1) = 6$.
3. Row 2 $(0, 2)$: $0 \cdot 3 + 2 \cdot (-1) = -2$.

$(6, -2)$ **– this matrix doubles every point's distance from the origin.**

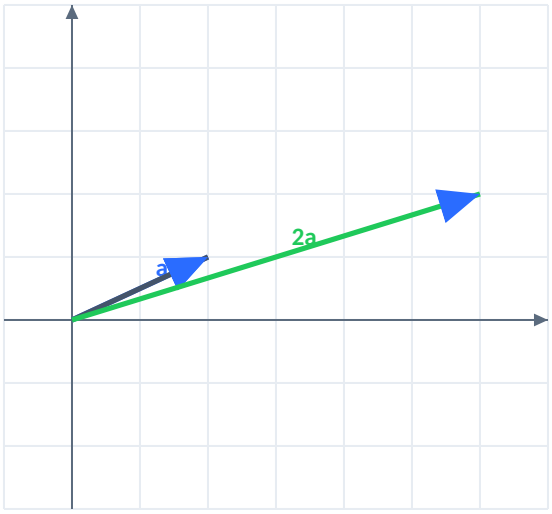
Example 8

Compute the combined vector $2\mathbf{a} - \mathbf{b}$ where $\mathbf{a} = (3, 1)$ and $\mathbf{b} = (4, -2)$.

SOLUTION

1. Do the stretch first: $2\mathbf{a} = (\, 2 \times 3\,,\ 2 \times 1\,) = (6, 2)$. *scale every part*
2. Subtracting a vector means subtracting matching components: $(6, 2) - (4, -2)$.
3. x: $6 - 4 = 2$. y: $2 - (-2) = 4$. *minus a minus → plus*

$2\mathbf{a} - \mathbf{b} = (2, 4)$.



Example 8: 2a (green) lies along the same line as a (blue) but reaches twice as far, to (6,2). Subtracting b then nudges the tip to the final answer (2,4).

Example 9

Find the magnitude of the 3-component vector $\mathbf{v} = (2, 3, 6)$.

SOLUTION

1. Magnitude in three dimensions extends Pythagoras with one more square: $|\mathbf{v}| = \sqrt{x^2 + y^2 + z^2}$.
   add the z^2 too
2. Square each component: $2^2 = 4$, $3^2 = 9$, $6^2 = 36$.
3. Add them: $4 + 9 + 36 = 49$.
4. Take the root: $\sqrt{49} = 7$. *perfect square*

$|\mathbf{v}| = 7$ – **a clean whole-number length.**

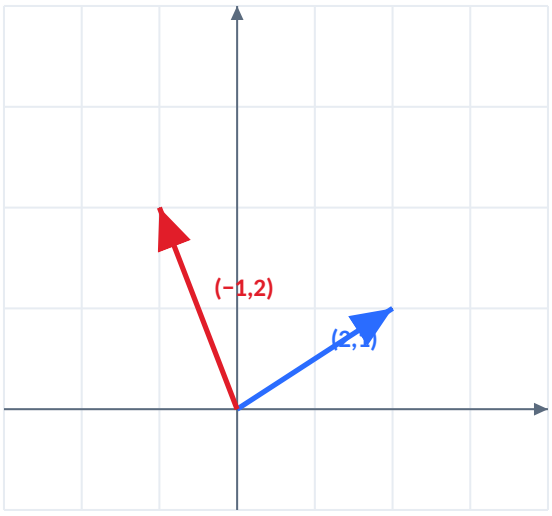
Example 10

The 90° rotation matrix is $\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$. Where does it send the point $(2, 1)$?

SOLUTION

1. Dot each row of the matrix with the column $\begin{bmatrix} 2 \\ 1 \end{bmatrix}$. *row · column*
2. Row 1 $(0, -1)$: $0 \cdot 2 + (-1) \cdot 1 = -1$.
3. Row 2 $(1, 0)$: $1 \cdot 2 + 0 \cdot 1 = 2$.
4. So $(2, 1)$ lands at $(-1, 2)$ – swapped and one sign flipped, exactly a quarter-turn anticlockwise.
   turn 90°

$(-1, 2)$ – **the matrix rotates every point 90° about the origin.**



Example 10: the blue point (2,1) is carried 90° anticlockwise to the red point (−1,2). A single matrix performs the whole rotation.

⚡ Mental-Math Shortcut

Scaling a vector or matrix? Just scale the parts. To compute $5\,(2, 3, 4)$ you do not think "vector" at all – multiply each number: $(10, 15, 20)$. Same for a grid. And to add, line the two up and add column by column in your head. Two quick checks that catch most slips: adding needs **equal sizes**, and a scalar touches **every** entry (it is easy to forget the zeros and negatives).

Why are we *allowed* to add vectors "slot by slot" and trust that everything still behaves like ordinary numbers? Because vector addition **inherits the laws of number addition**, one component at a time. This is a small but important logical argument: a property holds for the whole vector precisely because it holds in each slot.

Take the **commutative law**, $\mathbf{a} + \mathbf{b} = \mathbf{b} + \mathbf{a}$ (order of adding does not matter). Let $\mathbf{a} = (a_1, a_2)$ and $\mathbf{b} = (b_1, b_2)$. Then:

$$\mathbf{a} + \mathbf{b} = (a_1 + b_1,\ a_2 + b_2).$$

Now look inside each slot. The first slot is $a_1 + b_1$, and for plain numbers we *already know* $a_1 + b_1 = b_1 + a_1$. The second slot is $a_2 + b_2 = b_2 + a_2$ for the same reason. Substituting these back:

$$(a_1 + b_1,\ a_2 + b_2) = (b_1 + a_1,\ b_2 + a_2) = \mathbf{b} + \mathbf{a}.$$

So the two sums are equal **because each component was equal**. The reasoning pattern – "a claim about a structured object is true exactly when it is true in every part" – is how almost all vector and matrix laws are proved. (Matrix multiplication is the famous exception: it is *not* commutative, $AB \neq BA$ in general, precisely because its rule mixes rows with columns rather than acting slot by slot.)

⚠ Common Mistakes

- Adding vectors or matrices of **different sizes**. You can only add when the shapes match exactly, because addition pairs up matching slots.
- Forgetting to scale **every** entry. $3(2, -1, 0)$ is $(6, -3, 0)$ – the negative and the zero get multiplied too.
- Confusing the **magnitude** with adding the components. The length of $(3, 4)$ is $\sqrt{3^2 + 4^2} = 5$, not $3 + 4 = 7$.
- Reading a matrix entry in the wrong order. Entries are named **row first, then column**: in $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ the "row 2, column 1" entry is 3, not 2.
- Assuming matrix multiplication can be done in any order. AB and BA usually differ, so you cannot swap the factors the way you can with numbers.

Quiz

36.1.Q1. Add the vectors $(2, 3)$ and $(4, 1)$.

A. $(6, 4)$

B. $(8, 3)$

C. $(2, 4)$

D. $(6, 3)$

36.1.Q2. What is $4\,(1, -2)$?

36.1.Q3. Find the magnitude of the vector $(6, 8)$.

A. 14

B. 10

C. 48

D. 100

36.1.Q4. What size is the matrix $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$?

A. 3×2

B. 2×3

C. 2×2

D. 6×1

36.1.Q5. Add $\begin{bmatrix} 1 & 0 \\ 2 & 3 \end{bmatrix} + \begin{bmatrix} 4 & 1 \\ 0 & 5 \end{bmatrix}$.

36.1.Q6. True or False: you can add a 2×2 matrix to a 2×3 matrix.

A. True

B. False

36.1.Q7. Multiply $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ by the vector $(7, 5)$.

36.1.Q8. In $\begin{bmatrix} 9 & 8 \\ 7 & 6 \end{bmatrix}$, what is the entry in row 2, column 2?

A. 9

B. 8

C. 7

D. 6

36.1.Q9. True or False: for matrices, AB always equals BA.

A. True

B. False

🐍 From Concept to Code (Python)

Using only built-in lists, a **vector** is a flat list `[3, 1]` and a **matrix** is a list **of lists** `[[1, 2], [3, 4]]` (each inner list is one row). To add two vectors we walk the slots together with `zip` and add each pair – the slot-by-slot rule from the lesson, written in code. No libraries needed.

```
def add(u, v):
    return [a + b for a, b in zip(u, v)]   # add slot by slot

def scale(k, v):
    return [k * a for a in v]              # stretch every component

print(add([3, 1], [1, 2]))                 # (3,1)+(1,2)
print(scale(2, [2, 1]))                    # double the vector
```

```
output
[4, 3]
[4, 2]
```

```
# a matrix is a list of rows; print it row by row
M = [[1, 2],
     [3, 4]]
print(M[1][0])            # row 2, column 1  -> 3

# magnitude of a vector, straight from Pythagoras
import math
v = [3, 4]
print(math.hypot(*v))     # sqrt(3**2 + 4**2)
```

```
output
3
5.0
```

Try it:

1. Use `add` to compute $(5, 2, -1) + (0, 3, 4)$ on three-component vectors – the same function works unchanged for any length.
2. Write `matrix_add(A, B)` that adds two 2×2 matrices by adding matching rows with your `add` function, and test it on $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} + \begin{bmatrix} 5 & 6 \\ 7 & 8 \end{bmatrix}$.

Answer Key – Chapter 36 – Matrices & Vectors

36.1.Q1. $(6, 4)$ – add components: $2 + 4$ and $3 + 1$.

36.1.Q2. $(4, -8)$ – multiply each component by 4.

36.1.Q3. 10 – $\sqrt{6^2 + 8^2} = \sqrt{100} = 10$.

36.1.Q4. 2×3 – two rows, three columns (rows are stated first).

36.1.Q5. $\begin{bmatrix} 5 & 1 \\ 2 & 8 \end{bmatrix}$ – add entry by entry.

36.1.Q6. False – addition requires identical sizes.

36.1.Q7. $(7, 5)$ – the identity matrix leaves every vector unchanged.

36.1.Q8. 6 – bottom-right; rows then columns.

36.1.Q9. False – matrix multiplication is generally not commutative.



PART VIII

Trigonometry

The mathematics of angles, triangles and circular motion – and of every wave, rotation and oscillation. From right-triangle ratios to the unit circle and periodic functions.

Chapter 37 – Angles & Radians · Chapter 38 – Sine, Cosine & Tangent · Chapter 39 – The Unit Circle · Chapter 40 – Trigonometric Identities · Chapter 41 – Inverse Trigonometric Functions · Chapter 42 – Waves & Periodic Functions · Chapter 43 – Rotations & Polar Coordinates

Chapter 37 · Angles & Radians

> *How to measure an angle in two ways – degrees and radians – convert between them, and use the radian measure to read off arc length on a circle.*

The Idea Behind It

A full turn is exactly 2π radians (not some messy number). *but why?*

A **radian** is defined by the arc it cuts: an angle of one radian opens an arc whose length equals the **radius**. So "how many radians" really asks *how many radius-lengths of arc the angle sweeps*. The proof leans on one fact you already trust, the circumference of a circle:

$$\boxed{\text{circumference} = 2\pi r}$$

A full turn sweeps the **whole** rim, so the arc it cuts is the entire circumference $2\pi r$. Divide that arc by one radius r to count the radius-lengths:

$$\text{full turn} = \frac{\text{arc}}{\text{radius}} = \frac{2\pi r}{r} = 2\pi \text{ radians}$$

The r cancels cleanly, so the answer is the pure number 2π, true for a circle of *any* size. The same count gives a half turn $\pi r / r = \pi$:

$$\therefore \text{ full turn} = 2\pi \text{ rad} = 360^\circ, \qquad \text{half turn} = \pi \text{ rad} = 180^\circ$$

So 2π is not arbitrary at all: it is simply how many radius-lengths fit around the rim.

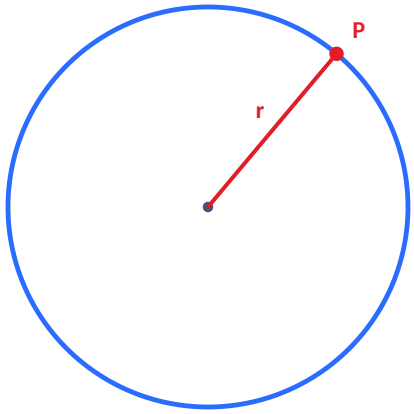
that is why π haunts every circle

An **angle** measures an amount of *turning*. Picture standing still and spinning on the spot: a quarter turn, a half turn, a full turn. To put numbers on that turning we need a unit, and there are two in common use. One you already know – **degrees**. The other – the **radian** – is the one every scientific calculator and every line of code secretly prefers. This chapter explains both and shows how to move between them.

Degrees – the everyday unit

Long ago the full turn was sliced into 360 equal parts, and each part was called one **degree**, written 1°. So a full turn is 360°, a half turn is 180° (a straight line), and a quarter turn is 90° (a square corner, a **right angle**). The number 360 is convenient because it divides evenly so many ways, but it is an arbitrary human choice – nothing in nature insists on it.

angle degree



An angle is swept anticlockwise from the positive x-axis (the three o-clock direction). Here the red radius has turned to point P.

Radians – measuring an angle by the arc it cuts

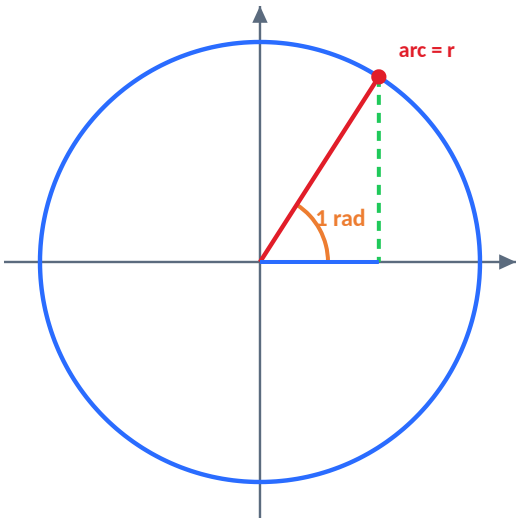
The radian takes a different, more natural idea. Draw a circle and pick an angle at its centre. That angle opens up an **arc** – a slice of the rim. The **radian** measure of the angle is simply *how many radius-lengths of arc it cuts off*.

> **RADIAN**
>
> An angle of **one radian** cuts off an arc whose length equals the circle's **radius**. So if the radius is r and the arc is also r long, the angle is exactly 1 radian.

Because the whole way around a circle (the **circumference**) has length $2\pi r$, and each radian uses up one radius r of that arc, a full turn must be 2π radians. That single fact unlocks everything:

$$\text{full turn} = 360^\circ = 2\pi \text{ rad}, \qquad \text{half turn} = 180^\circ = \pi \text{ rad}.$$

radian arc



One radian: the orange arc along the rim is exactly as long as the radius. One radian is about 57.3°.

Converting between degrees and radians

Everything flows from $180^\circ = \pi$ radians. Divide both sides and you get the two conversion factors. To go from degrees to radians, **multiply by** $\frac{\pi}{180}$; to go the other way, **multiply by** $\frac{180}{\pi}$.

$$\text{radians} = \text{degrees} \times \frac{\pi}{180}, \qquad \text{degrees} = \text{radians} \times \frac{180}{\pi}.$$

Note: A handy mental picture: π rad $= 180^\circ$ "halfway round", $\frac{\pi}{2}$ rad $= 90^\circ$ "a right angle", and 2π rad $= 360^\circ$ "all the way round".

Arc length – the radian pay-off

Here is why radians are worth the trouble. The length s of the arc cut by an angle θ is just the radius times the angle – **as long as** θ **is in radians**. No messy fractions of 360.

$$s = r\,\theta \qquad (\theta \text{ in radians}).$$

Worked examples

Example 1

Convert 90° to radians.

SOLUTION

1. Multiply the degrees by $\frac{\pi}{180}$: $90 \times \frac{\pi}{180}$. ×π/180
2. Simplify the fraction: $\frac{90}{180} = \frac{1}{2}$.
3. So the answer is $\frac{1}{2}\pi = \frac{\pi}{2}$.

$90^\circ = \frac{\pi}{2}$ **rad (a right angle is a quarter of 2π).**

Example 2

Convert 60° to radians.

SOLUTION

1. Multiply by $\frac{\pi}{180}$: $60 \times \frac{\pi}{180} = \frac{60\pi}{180}$. ×π/180
2. Simplify: $\frac{60}{180} = \frac{1}{3}$.

$60^\circ = \frac{\pi}{3}$ **rad.**

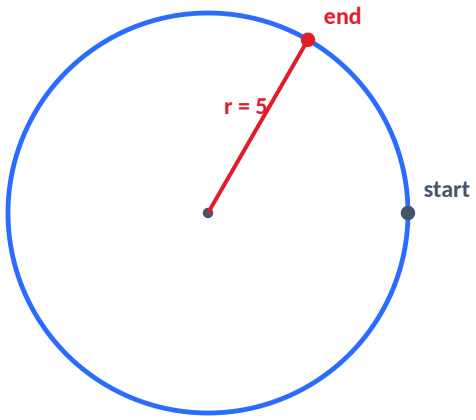
Example 3

Convert $\frac{\pi}{4}$ radians to degrees.

SOLUTION

1. Multiply the radians by $\frac{180}{\pi}$: $\frac{\pi}{4} \times \frac{180}{\pi}$. ×180/π
2. The π on top cancels the π underneath: $\frac{180}{4}$.
3. Divide: $\frac{180}{4} = 45$.

$\frac{\pi}{4}$ **rad** $= 45^\circ$**.**



Example 4: an angle of pi/3 rad (60 degrees) at the centre opens an arc from "start" to "end". Its length is s = r times theta = 5 times pi/3.

Example 4

A circle has radius $r = 5$ cm. Find the length of the arc cut by an angle of $\frac{\pi}{3}$ radians.

SOLUTION

1. The angle is already in radians, so use $s = r\theta$ directly. *already rad*
2. Substitute: $s = 5 \times \frac{\pi}{3} = \frac{5\pi}{3}$ cm.
3. As a decimal, $\frac{5\pi}{3} \approx 5.24$ cm.

The arc is $\frac{5\pi}{3} \approx 5.24$ cm long.

Example 5

A wheel of radius 0.40 m turns through 2 radians. How far does a point on its rim travel?

SOLUTION

1. Distance along the rim is arc length, so use $s = r\theta$.
2. The angle 2 is in radians, so no conversion is needed. *already rad*
3. Substitute: $s = 0.40 \times 2 = 0.80$ m.

A rim point travels 0.80 m.

Example 6

An arc of length 12 cm sits on a circle of radius 4 cm. What angle (in radians) does it subtend at the centre?

SOLUTION

1. Start from $s = r\theta$ and rearrange for θ: $\theta = \dfrac{s}{r}$. ÷ by r
2. Substitute: $\theta = \dfrac{12}{4} = 3$.
3. The units of length cancel, so the answer is a pure number of radians.

$\theta = 3$ **radians (a little under half a turn, since $\pi \approx 3.14$).**

Example 7

Convert 135° to radians, leaving the answer as a multiple of π.

SOLUTION

1. Multiply by $\dfrac{\pi}{180}$: $135 \times \dfrac{\pi}{180} = \dfrac{135\pi}{180}$. ×π/180
2. Simplify the fraction by dividing top and bottom by 45: $\dfrac{135}{180} = \dfrac{3}{4}$. ÷45 each

$135^\circ = \dfrac{3\pi}{4}$ **rad (three right-angles' worth, i.e. $\frac{3}{4}$ of a half-turn).**

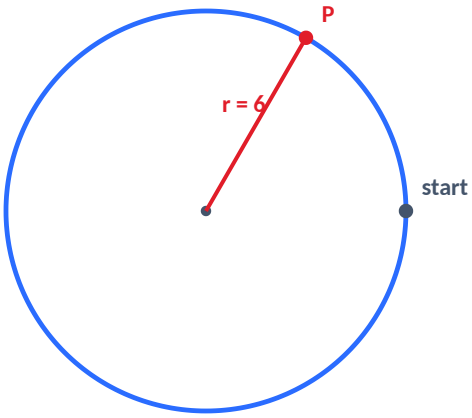
Example 8

A circular running track has radius 50 m. A runner sweeps through $\dfrac{3\pi}{2}$ radians around the centre. How far have they run along the track?

SOLUTION

1. The distance along the rim is arc length, and the angle is already in radians, so use $s = r\theta$. already rad
2. Substitute: $s = 50 \times \dfrac{3\pi}{2} = \dfrac{150\pi}{2} = 75\pi$ m. cancel the 2
3. As a decimal, $75\pi \approx 235.6$ m.

The runner covers $75\pi \approx 236$ m – three-quarters of the full lap.



Example 9: an arc of length 4 cm rests on a circle of radius 6 cm. The angle it opens at the centre is θ = s/r = 4/6 radians.

Example 9

An arc of length 4 cm lies on a circle of radius 6 cm. Find the angle it subtends, in radians and then in degrees.

SOLUTION

1. Rearrange $s = r\theta$ to $\theta = \frac{s}{r}$: $\theta = \frac{4}{6} = \frac{2}{3}$ rad. ÷ by r, then simplify
2. To convert to degrees, multiply by $\frac{180}{\pi}$: $\frac{2}{3} \times \frac{180}{\pi} = \frac{360}{3\pi} = \frac{120}{\pi}$.
3. As a decimal, $\frac{120}{\pi} \approx 38.2^\circ$. ÷ by π ≈ 3.14

$\theta = \frac{2}{3}$ **rad** $\approx 38.2^\circ$.

Note: Radians are "unit-free": an arc of 12 cm over a radius of 4 cm gives 3, with the centimetres cancelling. That is why calculators and code expect radians by default.

Mental-Math Shortcut

Memorise just the spine of the circle and read everything off it. $180^\circ = \pi$. Then $90^\circ = \frac{\pi}{2}$, $60^\circ = \frac{\pi}{3}$, $45^\circ = \frac{\pi}{4}$, $30^\circ = \frac{\pi}{6}$ – notice the degrees and the denominators multiply to 180 (60×3, 45×4, 30×6). To turn any "nice" degree value into radians, write it as a fraction of 180 and stick a π on top: $120^\circ = \frac{120}{180}\pi = \frac{2\pi}{3}$.

🧠 The Logic of It

Where does the magic formula $s = r\theta$ actually come from? Reason it out by **proportion**, no calculus required.

A full turn (2π radians) sweeps out the *entire* circumference, which is $2\pi r$. The arc length grows in exact step with the angle – double the angle, double the arc – so the ratio "arc to angle" is the same for any slice as it is for the whole circle:

$$\frac{s}{\theta} = \frac{2\pi r}{2\pi}.$$

The 2π on the right cancels, leaving $\frac{s}{\theta} = r$, and multiplying both sides by θ gives $s = r\theta$. The key logical step is *proportional reasoning*: because arc and angle increase together at a constant rate, one full-turn fact is enough to pin down every partial turn. This is exactly why the radian – defined so that the full turn is 2π – makes the formula so clean. Had we kept degrees, that 2π would have been a 360, and it would not cancel, so we would be stuck carrying a clumsy $\frac{\pi}{180}$ everywhere.

⚠ Common Mistakes

- Feeding **degrees** into a calculator or program that expects radians. $\sin(30)$ in radians is **not** 0.5 – you must first convert 30° to $\frac{\pi}{6}$ rad.
- Using $s = r\theta$ with θ in **degrees**. The formula only works when θ is in radians; convert first.
- Writing $360^\circ = \pi$ rad. It is $180^\circ = \pi$ rad; a *full* turn is 2π.
- Multiplying by the wrong factor: degrees$\to$radians uses $\frac{\pi}{180}$ (radians are smaller numbers), radians$\to$degrees uses $\frac{180}{\pi}$.
- Treating a radian as a length. A radian is an *angle*; the arc it cuts has a length, but the angle itself is just a number.

Quiz

37.1.Q1. How many radians are in a full turn?

A. π

B. 2π

C. 180

D. 360

37.1.Q2. Convert 180° to radians.

37.1.Q3. Convert 45° to radians.

A. $\frac{\pi}{2}$

B. $\frac{\pi}{3}$

C. $\frac{\pi}{4}$

D. $\frac{\pi}{6}$

37.1.Q4. Convert $\frac{\pi}{2}$ radians to degrees.

37.1.Q5. Roughly how many degrees is 1 radian?

A. About 57°

B. About 90°

C. About 100°

D. Exactly 60°

37.1.Q6. A circle has radius 3 m. Find the arc length cut by an angle of 2 radians.

37.1.Q7. An arc of 10 cm lies on a circle of radius 5 cm. What angle (radians) does it subtend?

37.1.Q8. True or False: the formula $s = r\theta$ works whether θ is in degrees or radians.

A. True

B. False

🐍 From Concept to Code (Python)

Python's `math` module thinks in **radians**, so it ships two converters – `math.radians(d)` (degrees in, radians out) and `math.degrees(r)` (the reverse) – plus the constant `math.pi`. Whenever you have an angle in degrees, convert it *before* doing any trigonometry.

```
import math

print(math.radians(180))      # 180 degrees as radians
print(math.degrees(math.pi))  # pi radians back to degrees

r, theta = 5, math.pi / 3     # radius 5, angle pi/3 rad
print(round(r * theta, 2))    # arc length s = r * theta
```

```
output
3.141592653589793
180.0
5.24
```

Try it:

1. Print `math.radians(90)` and confirm it equals `math.pi / 2`.
2. Write `arc(r, deg)` that takes the angle in **degrees**, converts it with `math.radians`, and returns the arc length `r * theta`. Test it with `r=4, deg=90`.

Answer Key – Chapter 37 – Angles & Radians

37.1.Q1. 2π – the circumference $2\pi r$ uses up 2π radius-lengths of arc.

37.1.Q2. π – by definition $180^\circ = \pi$ rad (a half turn).

37.1.Q3. $\dfrac{\pi}{4}$ – since $45 \times \dfrac{\pi}{180} = \dfrac{\pi}{4}$.

37.1.Q4. 90° – multiply by $\dfrac{180}{\pi}$ to get $\dfrac{180}{2} = 90$.

37.1.Q5. About 57° – since $\dfrac{180}{\pi} \approx 57.3$.

37.1.Q6. 6 m – use $s = r\theta = 3 \times 2$.

37.1.Q7. 2 radians – $\theta = \dfrac{s}{r} = \dfrac{10}{5}$.

37.1.Q8. False – it only holds when θ is in radians.

Chapter 38 · Sine, Cosine & Tangent

How to use the three trigonometric ratios – sine, cosine and tangent – to find an unknown side or angle in a right-angled triangle, using the memory aid SOH-CAH-TOA.

The Idea Behind It

A ratio like $\sin\theta = \frac{\text{opp}}{\text{hyp}}$ depends only on the angle, not on how big the triangle is. *but why?*

This is the fact that makes trigonometry possible: $\sin 30^\circ$ is one fixed number whether the triangle is tiny or enormous. The reason is **similar triangles**. Enlarge a right triangle by a scale factor k (a photocopy at k times the size). Every length is multiplied by the same k:

$$\boxed{\text{opp} \to k \cdot \text{opp}, \qquad \text{hyp} \to k \cdot \text{hyp}}$$

Now build the sine of the *enlarged* triangle and watch the scale factor cancel:

$$\sin\theta = \frac{k \cdot \text{opp}}{k \cdot \text{hyp}} = \frac{\text{opp}}{\text{hyp}}$$

The k on top divides out the k underneath, leaving the **same** ratio you started with. Nothing about the size survived, so the value can depend only on what the rescaling did *not* change, which is the angle θ:

$$\therefore\ \sin\theta \text{ is fixed by } \theta \text{ alone} \quad (\text{and likewise } \cos\theta,\ \tan\theta)$$

That is why a single table of values works for every right triangle on Earth, and why $\sin$, $\cos$ and $\tan$ are honest *functions of the angle*. *one angle, one ratio*

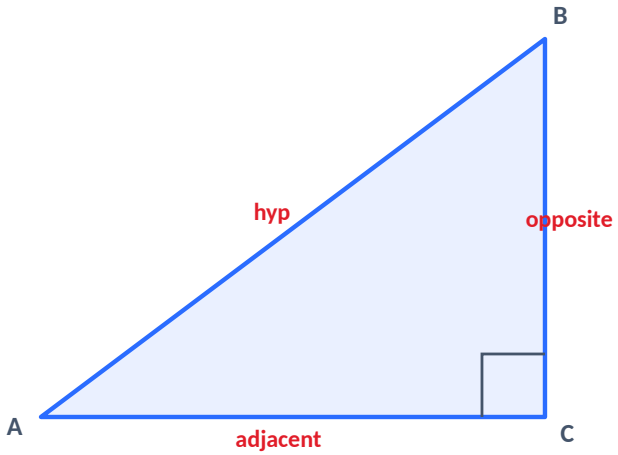
Trigonometry begins with a single, surprising fact about right-angled triangles: **if you know one of the non-right angles, the shape is fixed forever, no matter how big the triangle is.** Scale a 30° right triangle up or down and the *ratios* of its sides never change. Those fixed ratios have names – **sine**, **cosine** and **tangent** – and with them you can find a side you cannot reach with a ruler, or an angle you cannot measure with a protractor.

Naming the three sides

Stand inside the triangle at the angle you care about (call it θ). From there, the three sides have fixed names:

- The **hypotenuse** is the longest side, always opposite the right angle. It never moves.
- The **opposite** side is the one directly across from your angle θ.
- The **adjacent** side is the remaining one, touching θ (but not the hypotenuse).

hypotenuse opposite adjacent



Working from angle A: the side across from A is "opposite", the side along the bottom touching A is "adjacent", and the slope is the hypotenuse. The small square marks the right angle at C.

The three ratios – SOH-CAH-TOA

Each trig ratio compares two of those sides. The classic memory aid stacks the three definitions into one chant: **SOH-CAH-TOA**.

$$\sin\theta = \frac{\text{Opposite}}{\text{Hypotenuse}}, \quad \cos\theta = \frac{\text{Adjacent}}{\text{Hypotenuse}}, \quad \tan\theta = \frac{\text{Opposite}}{\text{Adjacent}}.$$

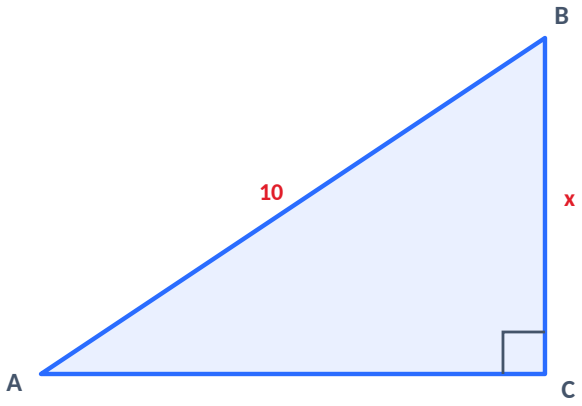
SOH-CAH-TOA

Three reminders in one: **S**in = **O**pp/**H**yp, **C**os = **A**dj/**H**yp, **T**an = **O**pp/**A**dj. Say it out loud once and you have all three ratios.

Note: Because tangent is opposite over adjacent, and sine and cosine share the hypotenuse, it is always true that $\tan\theta = \dfrac{\sin\theta}{\cos\theta}$ – the hypotenuse cancels.

Using a ratio to find a side

Pick the ratio that links the side you **know** to the side you **want**, write the equation, and solve. Below, angle A is known and we want the side opposite it; sine connects them.



Angle A = 30 degrees and the hypotenuse is 10. The opposite side x is found from sin A = x / 10.

Worked examples

Example 1

A right triangle has hypotenuse 10 and the angle $A = 30^\circ$. Find the side **opposite** A.

SOLUTION

1. The known side is the hypotenuse, the wanted side is the opposite – that is **SOH**, so use sine. *opp/hyp*
2. Write the ratio: $\sin 30^\circ = \dfrac{\text{opp}}{10}$.
3. Look up $\sin 30^\circ = 0.5$ and substitute: $0.5 = \dfrac{\text{opp}}{10}$.
4. Multiply both sides by 10: $\text{opp} = 0.5 \times 10 = 5$. *× both sides*

The opposite side is 5.

Example 2

In a right triangle, angle $\theta = 60^\circ$ and the **adjacent** side is 8. Find the hypotenuse.

SOLUTION

1. Adjacent and hypotenuse are linked by **CAH**, so use cosine. *adj/hyp*
2. Write it: $\cos 60^\circ = \dfrac{\text{adj}}{\text{hyp}} = \dfrac{8}{\text{hyp}}$.
3. Use $\cos 60^\circ = 0.5$: $0.5 = \dfrac{8}{\text{hyp}}$.
4. Rearrange: $\text{hyp} = \dfrac{8}{0.5} = 16$. *÷ flips it up*

The hypotenuse is 16.

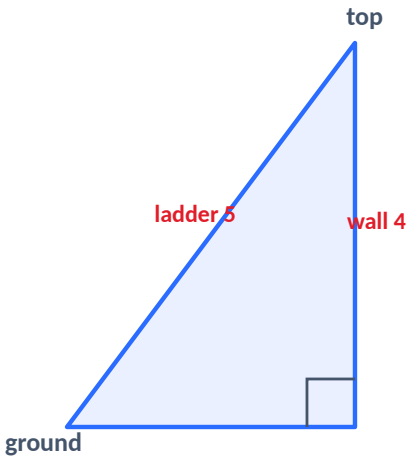
Example 3

A right triangle has the side opposite θ equal to 7 and the side adjacent to θ equal to 7. Find $\tan\theta$, then the angle θ.

SOLUTION

1. Opposite over adjacent is TOA, so use tangent: $\tan\theta = \frac{7}{7} = 1$. *opp/adj*
2. To recover the angle, ask "what angle has a tangent of 1?" – that is the inverse tangent, $\theta = \tan^{-1}(1)$. *run it backwards*
3. A 45° angle gives equal opposite and adjacent sides, so $\tan^{-1}(1) = 45^\circ$.

$\tan\theta = 1$ **and** $\theta = 45^\circ$**.**



Example 4: the ladder (hypotenuse, 5 m) leans from the ground up to a point 4 m high on the wall. The wall height is opposite the ground angle, so sin gives the angle.

Example 4

A ladder 5 m long leans against a wall, reaching 4 m up. What angle does it make with the ground?

SOLUTION

1. The ladder is the hypotenuse (5) and the height up the wall is opposite the ground angle (4).
2. Opposite and hypotenuse means sine: $\sin\theta = \frac{4}{5} = 0.8$. *SOH*
3. Take the inverse sine: $\theta = \sin^{-1}(0.8) \approx 53.1^\circ$. *angle from ratio*

The ladder leans at about 53.1° to the ground.

Example 5

A ramp rises at an angle of 15°. To gain a height of 2 m, how long must the ramp (the hypotenuse) be?

SOLUTION

1. Height is opposite the 15° angle, ramp length is the hypotenuse – sine again. SOH
2. Write it: $\sin 15^\circ = \dfrac{2}{\text{ramp}}$.
3. Use $\sin 15^\circ \approx 0.259$: $0.259 = \dfrac{2}{\text{ramp}}$.
4. Rearrange: $\text{ramp} = \dfrac{2}{0.259} \approx 7.7$ m. unknown on bottom → flip

The ramp must be about 7.7 m long.

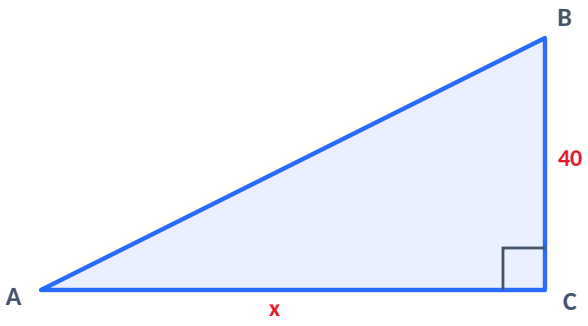
Example 6

A right triangle has legs 3 (adjacent) and 4 (opposite) about angle θ. Find all three ratios.

SOLUTION

1. First the hypotenuse, by Pythagoras: $\sqrt{3^2 + 4^2} = \sqrt{25} = 5$. need hyp first
2. $\sin\theta = \dfrac{\text{opp}}{\text{hyp}} = \dfrac{4}{5} = 0.8$. SOH
3. $\cos\theta = \dfrac{\text{adj}}{\text{hyp}} = \dfrac{3}{5} = 0.6$. CAH
4. $\tan\theta = \dfrac{\text{opp}}{\text{adj}} = \dfrac{4}{3} \approx 1.33$. TOA

$\sin\theta = 0.8$, $\cos\theta = 0.6$, $\tan\theta \approx 1.33$ – **the famous 3-4-5 triangle.**



Example 7: from angle A = 40 degrees, the height (opposite) is 40 m and the ground distance x is adjacent. Tangent links opposite to adjacent.

Example 7

From a point on the ground, a tower's top sits at an angle of elevation of 40°. The tower is 40 m tall. How far is the point from the base?

SOLUTION

1. The height (40, opposite the 40°) and the ground distance (adjacent) are linked by TOA, so use tangent.
   opp/adj
2. Write it: $\tan 40^\circ = \dfrac{\text{opp}}{\text{adj}} = \dfrac{40}{x}$.
3. Use $\tan 40^\circ \approx 0.839$: $0.839 = \dfrac{40}{x}$.
4. The unknown is on the bottom, so flip: $x = \dfrac{40}{0.839} \approx 47.7$ m. *unknown on bottom → flip*

The point is about 47.7 m from the base.

Example 8

A right triangle has hypotenuse 20 and one angle 35°. Find **both** unknown legs (opposite and adjacent to the 35°).

SOLUTION

1. The opposite leg pairs hypotenuse with sine: $\sin 35^\circ = \dfrac{\text{opp}}{20}$, so $\text{opp} = 20 \times \sin 35^\circ$. *SOH*
2. With $\sin 35^\circ \approx 0.574$: $\text{opp} = 20 \times 0.574 = 11.5$.
3. The adjacent leg pairs hypotenuse with cosine: $\text{adj} = 20 \times \cos 35^\circ$. *CAH*
4. With $\cos 35^\circ \approx 0.819$: $\text{adj} = 20 \times 0.819 = 16.4$.

Opposite ≈ 11.5, adjacent ≈ 16.4 (and indeed $11.5^2 + 16.4^2 \approx 20^2$).

Example 9

A guy-wire runs from the top of a 12 m mast to a ground anchor, making a 50° angle with the ground. Find the wire's length, then how far the anchor sits from the mast's foot.

SOLUTION

1. The mast height (12) is opposite the 50° angle, the wire is the hypotenuse – use sine. *SOH*
2. Length: $\sin 50^\circ = \dfrac{12}{\text{wire}}$, so $\text{wire} = \dfrac{12}{\sin 50^\circ} = \dfrac{12}{0.766} \approx 15.7$ m. *unknown on bottom → flip*
3. The ground distance is adjacent to the angle, so use tangent: $\tan 50^\circ = \dfrac{12}{\text{ground}}$. *TOA*
4. Flip: $\text{ground} = \dfrac{12}{\tan 50^\circ} = \dfrac{12}{1.19} \approx 10.1$ m.

The wire is about 15.7 m long and the anchor is about 10.1 m from the foot.

Note: On a calculator (or in code) make sure the angle mode is **degrees** when your angle is in degrees, or **radians** when it is in radians. Mixing them is the single most common trigonometry error.

⚡ Mental-Math Shortcut

Lock in the "special angles" so you never reach for a calculator on the common ones. Use the **1-2-3 over 2** trick: write $\sin$ of $0^\circ, 30^\circ, 45^\circ, 60^\circ, 90^\circ$ as $\frac{\sqrt{0}}{2}, \frac{\sqrt{1}}{2}, \frac{\sqrt{2}}{2}, \frac{\sqrt{3}}{2}, \frac{\sqrt{4}}{2}$ – i.e. $0, \frac{1}{2}, \frac{\sqrt{2}}{2}, \frac{\sqrt{3}}{2}, 1$. **Cosine is the same list read backwards** $(1, \frac{\sqrt{3}}{2}, \ldots, 0)$, because cosine of an angle equals sine of its complement. So $\cos 30^\circ = \sin 60^\circ = \frac{\sqrt{3}}{2}$.

🧠 The Logic of It

Why should a ratio of sides depend on the *angle alone*, and not on how big you draw the triangle? The answer is **similar triangles**, and the reasoning is clean.

Take a right triangle with angle θ. Now scale every length by some factor k – a photocopy enlargement. The opposite side becomes $k \cdot \text{opp}$ and the hypotenuse becomes $k \cdot \text{hyp}$. Form the sine:

$$\frac{k \cdot \text{opp}}{k \cdot \text{hyp}} = \frac{\text{opp}}{\text{hyp}}.$$

The scale factor k cancels top and bottom, leaving the *same* value. So **every** right triangle sharing the angle θ gives the identical ratio. The logical shape here is: a property that survives every rescaling must depend only on what rescaling does *not* change – and what is unchanged is the angle. That is precisely why $\sin\theta$, $\cos\theta$ and $\tan\theta$ are well-defined functions of the angle, the bedrock the whole subject is built on.

⚠ Common Mistakes

- Mislabelling the sides. "Opposite" and "adjacent" are named *relative to the angle you chose*; switch angles and the two swap.
- Putting the ratio upside down. $\sin\theta$ is opposite **over** hypotenuse – the hypotenuse is on the bottom because it is the longest side, so $\sin$ and $\cos$ are always at most 1.
- Leaving the calculator in radian mode while using degree angles (or vice versa). $\sin(30)$ in radian mode is about -0.99, not 0.5.
- Using a trig ratio in a triangle that has no right angle. SOH-CAH-TOA only applies to right-angled triangles.
- Forgetting to take the inverse function when you want the *angle*. From $\sin\theta = 0.5$ you get θ by $\sin^{-1}(0.5) = 30^\circ$, not by dividing.

Quiz

38.1.Q1. Which ratio is opposite over hypotenuse?

A. $\sin$

B. $\cos$

C. $\tan$

D. none

38.1.Q2. What does CAH stand for?

38.1.Q3. If the opposite side is 3 and the hypotenuse is 6, what is $\sin\theta$?

A. 0.5

B. 2

C. 0.6

D. 1.5

38.1.Q4. A right triangle has adjacent $= 5$ and opposite $= 5$. What is $\tan\theta$?

38.1.Q5. Find the hypotenuse if $\cos 60^\circ = 0.5$ and the adjacent side is 4.

38.1.Q6. True or False: $\tan\theta = \dfrac{\sin\theta}{\cos\theta}$.

A. True

B. False

38.1.Q7. For the 3-4-5 triangle, what is $\cos\theta$ if 4 is opposite and 3 is adjacent to θ?

A. 0.6

B. 0.8

C. 1.33

D. 0.75

38.1.Q8. You know $\sin\theta = 0.5$. How do you find the angle θ?

🐍 From Concept to Code (Python)

Python's `math` module has `sin`, `cos` and `tan` – but they expect the angle in **radians**, so convert any degree angle with `math.radians` first. To go the other way (a ratio back to an angle) use `math.asin`, `math.acos`, `math.atan`, which return radians you can turn back with `math.degrees`.

```
import math

theta = math.radians(30)        # 30 degrees -> radians FIRST
print(round(math.sin(theta), 3))  # opposite / hypotenuse
print(round(math.cos(theta), 3))

opp, hyp = 4, 5                 # find the angle from a ratio
print(round(math.degrees(math.asin(opp / hyp)), 1))
```

```
output
0.5
0.866
53.1
```

Try it:

1. Print `math.tan(math.radians(45))` and confirm it is (very close to) `1`.
2. Write `opposite(hyp, deg)` that returns `hyp * sin(angle)` for an angle given in **degrees**. Test it with `hyp=10, deg=30` and check you get `5`.

Answer Key – Chapter 38 – Sine, Cosine & Tangent

38.1.Q1. $\sin$ – SOH: Sine = Opposite / Hypotenuse.

38.1.Q2. Cosine = Adjacent / Hypotenuse.

38.1.Q3. 0.5 – $\sin\theta = \dfrac{3}{6} = 0.5$.

38.1.Q4. 1 – $\tan\theta = \dfrac{5}{5} = 1$ (so $\theta = 45^\circ$).

38.1.Q5. 8 – $0.5 = \dfrac{4}{\text{hyp}}$ gives $\text{hyp} = \dfrac{4}{0.5} = 8$.

38.1.Q6. True – opp/hyp divided by adj/hyp cancels the hypotenuse to leave opp/adj.

38.1.Q7. 0.6 – $\cos\theta = \dfrac{\text{adj}}{\text{hyp}} = \dfrac{3}{5} = 0.6$.

38.1.Q8. Take the inverse sine: $\theta = \sin^{-1}(0.5) = 30^\circ$.

Chapter 39 · The Unit Circle

*How the unit circle extends sine and cosine to **any** angle – even past 90° and into the negatives – by reading them straight off the coordinates of a point on a circle of radius 1.*

The Idea Behind It

For every angle, $\sin^2\theta + \cos^2\theta = 1$. *but why?*

On the unit circle a point at angle θ sits at $(\cos\theta,\ \sin\theta)$: the horizontal coordinate **is** the cosine, the vertical one **is** the sine. The proof is just **Pythagoras**, $a^2 + b^2 = c^2$, applied to the little right triangle that drops from the point down to the x-axis:

$$\boxed{(\text{horizontal})^2 + (\text{vertical})^2 = (\text{radius})^2}$$

The two legs are $\cos\theta$ and $\sin\theta$, and the hypotenuse is the radius, which on the *unit* circle is exactly 1. Substitute those three lengths in:

$$(\ \cos\theta\ )^2 + (\ \sin\theta\ )^2 = \ 1\ ^2$$

Writing $(\cos\theta)^2$ as $\cos^2\theta$ and squaring the 1 on the right gives the famous identity:

$$\therefore\ \ \sin^2\theta + \cos^2\theta = 1$$

We assumed nothing about θ except that its point lives on the circle, so this holds for *every* angle at once.

it is Pythagoras in disguise

In the last chapter, sine and cosine lived inside right triangles, so they only made sense for angles between 0° and 90°. But a wheel turns past a quarter, a pendulum swings both ways, a signal oscillates forever. To handle *every* angle we move from the triangle to the **unit circle** – and the ideas suddenly become both simpler and far more powerful.

UNIT CIRCLE

The circle of radius **1** centred at the origin $(0, 0)$. Its equation is $x^2 + y^2 = 1$.

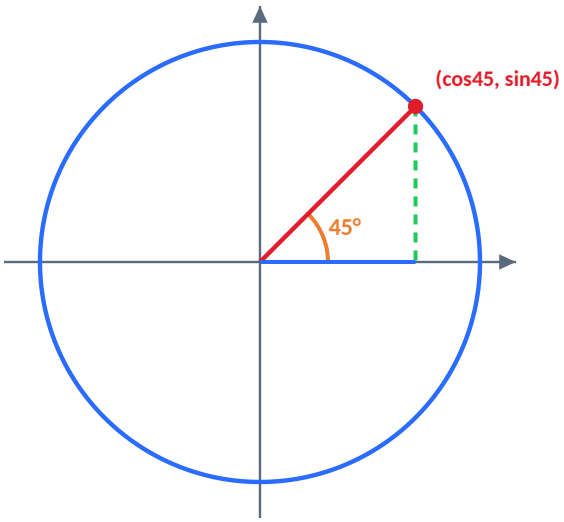
Cosine and sine as coordinates

Start a point at $(1, 0)$ – the rightmost edge – and rotate it anticlockwise by an angle θ. It traces along the rim and lands somewhere. The new position has an x-coordinate and a y-coordinate, and here is the whole secret of trigonometry:

$$\text{the point is } (\cos\theta,\ \sin\theta).$$

That is, the horizontal coordinate **is** the cosine, and the vertical coordinate **is** the sine. Why does this match the triangle definitions? Because the hypotenuse is the radius, which is 1. So $\cos\theta = \dfrac{\text{adj}}{1} = \text{adj} = x$, and $\sin\theta = \dfrac{\text{opp}}{1} = \text{opp} = y$. Dividing by 1 changes nothing, so the coordinates and the ratios are the same number.

unit circle quadrant



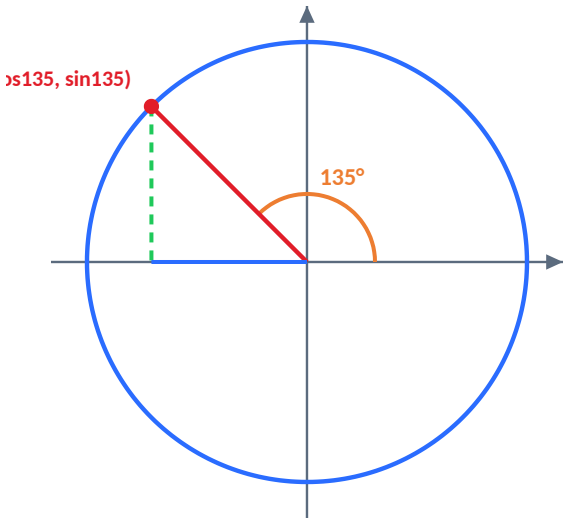
A point rotated 45 degrees round the unit circle. The blue base is cos (the x-coordinate); the green height is sin (the y-coordinate).

Signs by quadrant

Because cosine is the x-coordinate and sine is the y-coordinate, their **signs** are decided purely by *which quarter of the plane the point is in*. To the right, $x > 0$, so cosine is positive; above the axis, $y > 0$, so sine is positive.

Quadrant	Angle range	$\cos\theta$ (x)	$\sin\theta$ (y)
I	0° to 90°	$+$	$+$
II	90° to 180°	$-$	$+$
III	180° to 270°	$-$	$-$
IV	270° to 360°	$+$	$-$

A popular reminder for *which ratio is positive where* is **"All Students Take Calculus"**, reading the quadrants I, II, III, IV: **A**ll are positive in I, only **S**ine in II, only **T**angent in III, only **C**osine in IV.



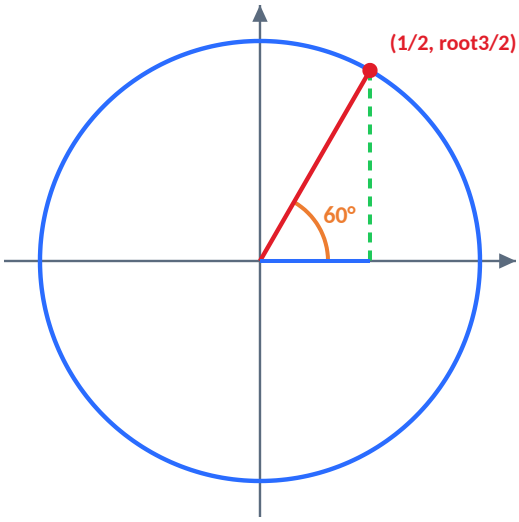
In Quadrant II (135 degrees) the point sits up and to the LEFT: x is negative so cos is negative, y is positive so sin is positive.

Special angles worth memorising

Four angles in the first quadrant come up constantly. Their exact coordinates are:

$$30^\circ : \left(\frac{\sqrt{3}}{2}, \frac{1}{2}\right) \quad 45^\circ : \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\right) \quad 60^\circ : \left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right)$$

At the four "corners" the values are even simpler: 0° is $(1, 0)$, 90° is $(0, 1)$, 180° is $(-1, 0)$, and 270° is $(0, -1)$.



The 60 degree point: cos 60 = 1/2 (short base), sin 60 = root3/2 which is about 0.87 (tall height).

Worked examples

Example 1

What are the coordinates of the point at 0° on the unit circle? What does this say about $\cos 0^\circ$ and $\sin 0^\circ$?

SOLUTION

1. At 0° the point has not moved from its start at $(1, 0)$. *no turn yet*
2. The x-coordinate is the cosine, the y-coordinate is the sine. *x=cos, y=sin*

$(1, 0)$, **so** $\cos 0^\circ = 1$ **and** $\sin 0^\circ = 0$.

Example 2

Find $\cos 90^\circ$ and $\sin 90^\circ$ from the circle.

SOLUTION

1. A 90° turn lands the point straight up at the top of the circle. *quarter turn*
2. The top of a radius-1 circle is the coordinate $(0, 1)$.
3. Read off: $x = 0$ is the cosine, $y = 1$ is the sine.

$\cos 90^\circ = 0$ **and** $\sin 90^\circ = 1$.

Example 3

In which quadrant is 200°, and what are the signs of $\cos 200^\circ$ and $\sin 200^\circ$?

SOLUTION

1. 200° is between 180° and 270°, so it is in **Quadrant III**. *past 180, before 270*
2. In Quadrant III the point is down and to the left: $x < 0$ and $y < 0$. *down-left → both −*

Quadrant III; both $\cos 200^\circ < 0$ **and** $\sin 200^\circ < 0$ **(negative).**

Example 4

Without a calculator, decide the sign of $\cos 300^\circ$.

SOLUTION

1. 300° is between 270° and 360°, so it is in **Quadrant IV**. *last quarter*
2. In Quadrant IV the point is below the axis but to the right, so $x > 0$. *right → cos +*
3. Cosine is the x-coordinate.

$\cos 300^\circ$ **is** positive.

Example 5

Use the special-angle values to write $\cos 60^\circ$ and $\sin 60^\circ$.

SOLUTION

1. The 60° point has coordinates $\left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right)$. *memorised value*
2. Cosine is the x-coordinate; sine is the y-coordinate. *x=cos, y=sin*

$\cos 60^\circ = \frac{1}{2}$ **and** $\sin 60^\circ = \frac{\sqrt{3}}{2} \approx 0.87$**.**

Example 6

A point on the unit circle is at $(\cos\theta, \sin\theta)$ with $\cos\theta = -\frac{1}{2}$ and $\sin\theta = +\frac{\sqrt{3}}{2}$. Which quadrant, and what is θ?

SOLUTION

1. $x < 0$ and $y > 0$, so the point is up-and-left: **Quadrant II**. *−x, +y → Q2*
2. The size of the coordinates matches the 60° point, but reflected into Quadrant II. *reference 60°*
3. The Quadrant II angle with this reference is $180^\circ - 60^\circ = 120^\circ$.

Quadrant II, and $\theta = 120^\circ$**.**

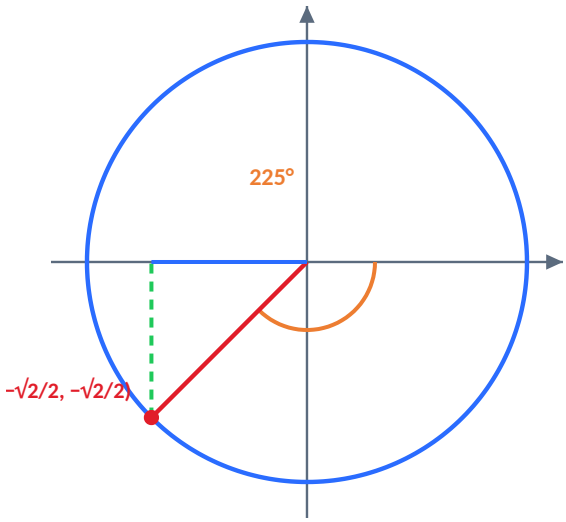
Example 7

Read $\cos 270^\circ$ and $\sin 270^\circ$ straight off the circle.

SOLUTION

1. A 270° turn is three-quarters of the way round, landing the point at the very bottom. *3/4 turn*
2. The bottom of a radius-1 circle is the coordinate $(0, -1)$.
3. Read off: $x = 0$ is the cosine, $y = -1$ is the sine. *x=cos, y=sin*

$\cos 270^\circ = 0$ **and** $\sin 270^\circ = -1$**.**



Example 8: the 225 degree point sits down-and-left in Quadrant III. Its reference angle off the horizontal is 225 − 180 = 45 degrees, so both coordinates are −√2/2.

Example 8

Find the exact coordinates of the point at 225°, and hence $\cos 225^\circ$ and $\sin 225^\circ$.

SOLUTION

1. 225° is between 180° and 270°, so it is in Quadrant III – down and to the left, so $x < 0$ and $y < 0$.

   both −

2. The reference angle is $225^\circ - 180^\circ = 45^\circ$, whose coordinates have size $\frac{\sqrt{2}}{2}$ each. reference 45°

3. Attach the Quadrant III signs (both negative): the point is $\left(-\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}\right)$.

$\cos 225^\circ = -\frac{\sqrt{2}}{2}$ **and** $\sin 225^\circ = -\frac{\sqrt{2}}{2}$ **(about** -0.71 **each).**

Example 9

Using $\tan\theta = \dfrac{\sin\theta}{\cos\theta}$, find $\tan 120^\circ$ from the unit-circle coordinates.

SOLUTION

1. The 120° point is in Quadrant II with coordinates $\left(-\frac{1}{2}, \frac{\sqrt{3}}{2}\right)$ (reference 60°, cos negative, sin positive).

   Q2: −x, +y

2. Tangent is the y-coordinate over the x-coordinate: $\tan 120^\circ = \dfrac{\sin 120^\circ}{\cos 120^\circ} = \dfrac{\frac{\sqrt{3}}{2}}{-\frac{1}{2}}$.

3. Dividing by $-\frac{1}{2}$ is the same as multiplying by -2: $\dfrac{\sqrt{3}}{2} \times (-2) = -\sqrt{3}$. ÷ ½ → × 2

$\tan 120^\circ = -\sqrt{3} \approx -1.73$ **– negative, as tangent always is in Quadrant II.**

Note: Sine and cosine never leave the range -1 to 1, because the point lives on a circle of radius 1 – no coordinate can exceed the radius. If you ever compute $\sin\theta = 1.4$, something has gone wrong.

⚡ Mental-Math Shortcut

To get sine or cosine of *any* angle, reduce it to a first-quadrant **reference angle** and then fix the sign by the quadrant. For 150°: it is $180^\circ - 150^\circ = 30^\circ$ from the horizontal, and it sits in Quadrant II where cosine is negative and sine positive. So $\cos 150^\circ = -\cos 30^\circ = -\frac{\sqrt{3}}{2}$ and $\sin 150^\circ = +\sin 30^\circ = +\frac{1}{2}$. **Reference angle for the number, quadrant for the sign** – that two-step turns the whole circle into the first quarter you already know.

🧠 The Logic of It

Why is it *legitimate* to call the coordinate definition the "same" sine and cosine as the triangle ones, rather than a brand-new pair that just happens to share a name? The argument is a careful **case check**.

In Quadrant I the point $(\cos\theta, \sin\theta)$ forms a right triangle with the axis whose hypotenuse is the radius 1. There, $\cos\theta = \dfrac{\text{adj}}{1} = x$ and $\sin\theta = \dfrac{\text{opp}}{1} = y$, so the two definitions *agree exactly* on the angles where the triangle definition was valid (0° to 90°). For angles outside that range there is no right triangle at all, so the old definition simply says nothing – there is nothing to contradict. We are therefore free to **extend** sine and cosine by the coordinate rule, and that extension is the unique one that (a) matches the old values where they existed and (b) keeps the point on the circle. The logical principle is *consistent extension*: a new definition that reproduces the old one on its whole domain, and fills in the gaps without conflict, is the honest continuation of the same idea – not a different function wearing its name.

⚠ Common Mistakes

- Swapping the coordinates: cosine is the x (horizontal) coordinate and sine is the y (vertical) one, not the other way round.
- Forgetting the sign. $\cos 200^\circ$ is **negative** because the point is in Quadrant III – taking just the reference value $\cos 20^\circ$ and leaving it positive is wrong.
- Believing sine or cosine can exceed 1. The point lies on a radius-1 circle, so both stay between -1 and 1.
- Mixing up the special coordinates of 30° and 60°. At 30° the height (sine) is the *small* one, $\frac{1}{2}$; at 60° the height is the *tall* one, $\frac{\sqrt{3}}{2}$.
- Measuring the angle clockwise. The standard convention is **anticlockwise** from the positive x-axis.

Quiz

39.1.Q1. On the unit circle, which coordinate gives $\cos\theta$?

A. the x-coordinate

B. the y-coordinate

C. the radius

D. the angle

39.1.Q2. What are the coordinates of the point at 180°?

39.1.Q3. In Quadrant II, what are the signs of cosine and sine?

A. cos $-$, sin $+$

B. cos $+$, sin $+$

C. cos $-$, sin $-$

D. cos $+$, sin $-$

39.1.Q4. Is $\sin 300^\circ$ positive or negative?

39.1.Q5. What is $\sin 90^\circ$?

A. 0

B. 1

C. -1

D. $\frac{1}{2}$

39.1.Q6. A point on the unit circle has $\sin\theta = 1.2$. What is wrong?

39.1.Q7. True or False: $\cos 60^\circ = \frac{1}{2}$.

A. True

B. False

39.1.Q8. Which quadrant contains the angle 250°?

🐍 From Concept to Code (Python)

You can walk the unit circle in code. Remember `math.cos` and `math.sin` take **radians**, so convert each degree angle with `math.radians`. Plotting $(\cos\theta, \sin\theta)$ for a few angles shows the signs flipping quadrant by quadrant.

```
import math

for deg in [0, 90, 180, 270]:
    t = math.radians(deg)
    x = round(math.cos(t), 2)
    y = round(math.sin(t), 2)
    print(deg, "->", (x, y))
```

```
output
0 -> (1.0, 0.0)
90 -> (0.0, 1.0)
180 -> (-1.0, 0.0)
270 -> (-0.0, -1.0)
```

Try it:

1. Print the coordinates for `deg = 135`. Check the x (cos) is negative and the y (sin) is positive – Quadrant II.
2. Loop over `range(0, 360, 45)` and confirm that $x^2 + y^2$ rounds to `1.0` for every angle (the point always sits on the unit circle).

Answer Key – Chapter 39 – The Unit Circle

39.1.Q1. The x-coordinate – a point at angle θ is $(\cos\theta, \sin\theta)$.

39.1.Q2. $(-1, 0)$, so $\cos 180^\circ = -1$ and $\sin 180^\circ = 0$.

39.1.Q3. cos negative, sin positive – the point is up and to the left.

39.1.Q4. Negative – 300° is in Quadrant IV, where the point is below the axis ($y < 0$).

39.1.Q5. 1 – the 90° point is at the top, $(0, 1)$.

39.1.Q6. Impossible – sine stays between -1 and 1 on a radius-1 circle.

39.1.Q7. True – the 60° point is $\left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right)$, and cosine is the x-coordinate.

39.1.Q8. Quadrant III – it lies between 180° and 270°.

Chapter 40 · Trigonometric Identities

*What a trigonometric **identity** is – an equation true for every angle – and how to use the master identity $\sin^2\theta + \cos^2\theta = 1$ to simplify expressions and find one ratio from another.*

The Idea Behind It

A whole second identity, $1 + \tan^2\theta = \sec^2\theta$, falls out of $\sin^2\theta + \cos^2\theta = 1$ for free. *but why?*

First, why is $\tan\theta = \dfrac{\sin\theta}{\cos\theta}$? Because $\tan = \dfrac{\text{opp}}{\text{adj}}$, and dividing top and bottom by the hypotenuse turns each side into a sine or cosine, the hypotenuse cancelling: $\dfrac{\text{opp}/\text{hyp}}{\text{adj}/\text{hyp}} = \dfrac{\sin\theta}{\cos\theta}$. Now take the master identity and do **one move**, divide every term by $\cos^2\theta$:

$$\boxed{\sin^2\theta + \cos^2\theta = 1}$$

$$\frac{\sin^2\theta}{\cos^2\theta} + \frac{\cos^2\theta}{\cos^2\theta} = \frac{1}{\cos^2\theta}$$

The first term is $\left(\dfrac{\sin\theta}{\cos\theta}\right)^2 = \tan^2\theta$; the middle term is $\dfrac{\cos^2\theta}{\cos^2\theta} = 1$; and $\dfrac{1}{\cos^2\theta}$ is written $\sec^2\theta$. Swap those names in:

$$\therefore \quad \tan^2\theta + 1 = \sec^2\theta$$

No new geometry was needed: one Pythagorean identity, divided by $\cos^2\theta$, hands you a second one (valid wherever $\cos\theta \neq 0$). *divide once, get an identity free*

Some equations are true only for special values: $x + 3 = 5$ holds *only* when $x = 2$. Other equations are true for **every** value you could put in: $2(x + 1) = 2x + 2$ holds no matter what x is. That second kind is called an **identity**, and trigonometry is full of them. They are the grammar of the subject – the rules that let you rewrite a tangled expression into a simple one.

IDENTITY

An equation that is true for **all** allowed values of the variable, not just a few. We often signal it with $\equiv$, but an ordinary $=$ is common too.

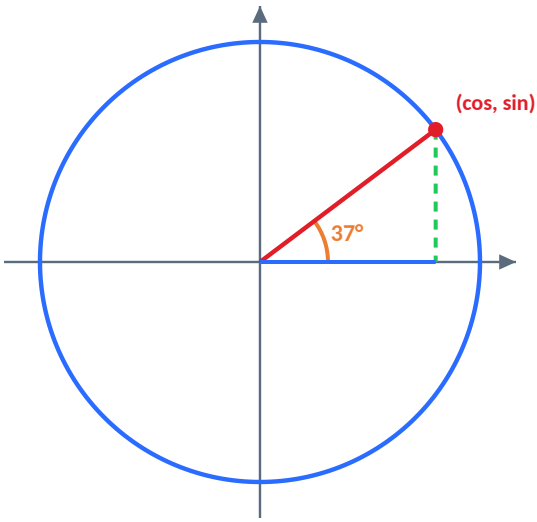
identity

The Pythagorean identity

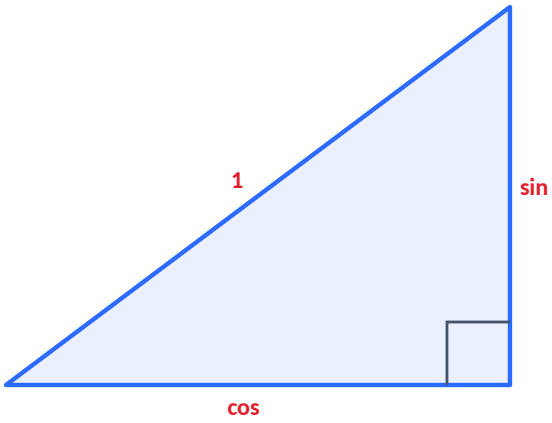
The most important trig identity comes straight from the unit circle. A point at angle θ has coordinates $(\cos\theta, \sin\theta)$, and it sits on a circle of radius 1. The equation of that circle is $x^2 + y^2 = 1$. Substitute $x = \cos\theta$ and $y = \sin\theta$:

$$\cos^2\theta + \sin^2\theta = 1.$$

(The notation $\cos^2\theta$ is shorthand for $(\cos\theta)^2$ – square the cosine.) This is the **Pythagorean identity**, so named because it is really the theorem $a^2 + b^2 = c^2$ applied to the reference triangle, whose legs are $\cos\theta$ and $\sin\theta$ and whose hypotenuse is the radius 1.



The reference triangle inside the unit circle: blue leg = cos, green leg = sin, red hypotenuse = 1. Pythagoras gives cos squared plus sin squared = 1 for every angle.



The same triangle pulled out flat: the legs cos and sin, hypotenuse 1. Pythagoras: cos squared + sin squared = 1 squared = 1.

Pythagorean identity

A second identity you already met

From Chapter 38, tangent is opposite over adjacent, which is sine over cosine:

$$\tan\theta = \frac{\sin\theta}{\cos\theta}.$$

This too is an identity – true for every angle where $\cos\theta \neq 0$ (we must avoid dividing by zero, which is why $\tan 90^\circ$ is undefined).

Rearranging the master identity

Because $\cos^2\theta + \sin^2\theta = 1$ holds always, you can move terms across to get two more useful forms. These let you find one ratio when you know the other:

$$\sin^2\theta = 1 - \cos^2\theta, \qquad \cos^2\theta = 1 - \sin^2\theta.$$

Note: When you take the square root to get $\sin\theta$ from $\sin^2\theta$, you must decide the **sign** yourself from the quadrant (Chapter 39). The identity gives the size; the quadrant gives the sign.

Worked examples

Example 1

Verify the identity $\cos^2\theta + \sin^2\theta = 1$ at $\theta = 30^\circ$.

SOLUTION

1. Recall the values: $\cos 30^\circ = \frac{\sqrt{3}}{2}$ and $\sin 30^\circ = \frac{1}{2}$. *plug in 30°*
2. Square them: $\cos^2 30^\circ = \frac{3}{4}$ and $\sin^2 30^\circ = \frac{1}{4}$.
3. Add: $\frac{3}{4} + \frac{1}{4} = 1$. ✓ *sums to 1*

It checks out: $\frac{3}{4} + \frac{1}{4} = 1$.

Example 2

Given $\sin\theta = \frac{3}{5}$ and that θ is in Quadrant I, find $\cos\theta$.

SOLUTION

1. Use $\cos^2\theta = 1 - \sin^2\theta$. *rearranged identity*
2. Substitute: $\cos^2\theta = 1 - \left(\frac{3}{5}\right)^2 = 1 - \frac{9}{25} = \frac{16}{25}$.
3. Square-root: $\cos\theta = \pm\frac{4}{5}$. In Quadrant I cosine is positive. *Q1: cos +*

$\cos\theta = \frac{4}{5}$ **(the 3-4-5 triangle again).**

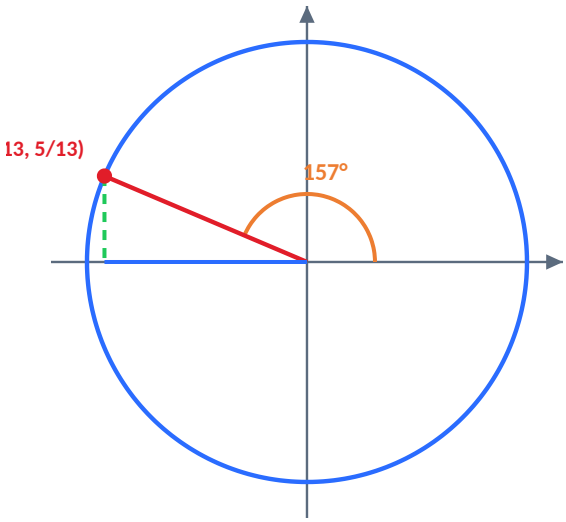
Example 3

Given $\cos\theta = -\dfrac{12}{13}$ with θ in Quadrant II, find $\sin\theta$.

SOLUTION

1. Use $\sin^2\theta = 1 - \cos^2\theta = 1 - \dfrac{144}{169} = \dfrac{25}{169}$. square cancels the −
2. Square-root: $\sin\theta = \pm\dfrac{5}{13}$.
3. In Quadrant II, sine is **positive**, so take the plus. Q2: sin +

$\sin\theta = \dfrac{5}{13}$.



Example 3 pictured: in Quadrant II the point is up and to the left, so cos (x) is negative but sin (y) is positive. The identity fixes the size 5/13; the quadrant fixes the plus sign.

Example 4

Simplify the expression $\dfrac{\sin^2\theta + \cos^2\theta}{\cos\theta}$.

SOLUTION

1. The numerator is the Pythagorean identity: $\sin^2\theta + \cos^2\theta = 1$. $\sin^2+\cos^2=1$
2. So the whole expression becomes $\dfrac{1}{\cos\theta}$.

$\dfrac{1}{\cos\theta}$ – **a single clean term instead of three.**

Example 5

Show that $\frac{\sin\theta}{\cos\theta} \cdot \cos\theta$ simplifies to $\sin\theta$.

SOLUTION

1. Replace $\frac{\sin\theta}{\cos\theta}$ with $\tan\theta$ if you like, or work directly. *tan = sin/cos*
2. The $\cos\theta$ in the denominator cancels the $\cos\theta$ being multiplied: $\frac{\sin\theta}{\cos\theta} \cdot \cos\theta$. *cos ÷ cos = 1*

It leaves $\sin\theta$.

Example 6

Prove the identity $(1 - \sin\theta)(1 + \sin\theta) = \cos^2\theta$.

SOLUTION

1. Expand the left side as a difference of squares: $(1 - \sin\theta)(1 + \sin\theta) = 1 - \sin^2\theta$.

   difference of squares
2. Now use the rearranged Pythagorean identity: $1 - \sin^2\theta = \cos^2\theta$. *1−sin²=cos²*
3. Left side equals right side, so the identity holds.

Proven: $(1 - \sin\theta)(1 + \sin\theta) = 1 - \sin^2\theta = \cos^2\theta$.

Example 7

Given $\sin\theta = \frac{5}{13}$ and $\cos\theta = \frac{12}{13}$ (Quadrant I), find $\tan\theta$.

SOLUTION

1. Use the quotient identity $\tan\theta = \frac{\sin\theta}{\cos\theta}$. *tan = sin/cos*
2. Substitute the two ratios: $\tan\theta = \frac{5/13}{12/13}$.
3. Dividing fractions, the common denominator 13 cancels: $\frac{5}{13} \times \frac{13}{12} = \frac{5}{12}$. *13 cancels*

$\tan\theta = \frac{5}{12}$ **– the 5-12-13 right triangle.**

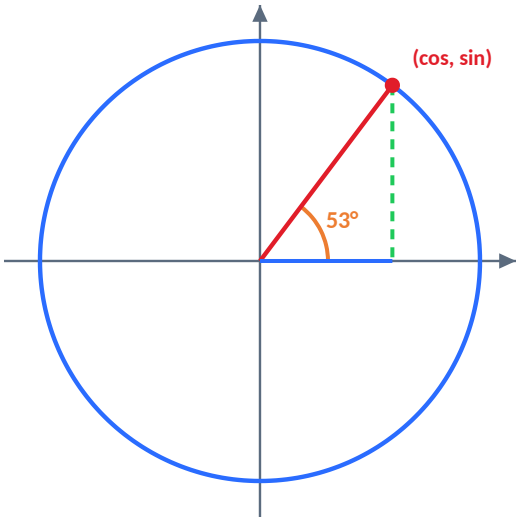
Example 8

Simplify $\dfrac{1-\cos^2\theta}{\sin\theta}$ to a single ratio.

SOLUTION

1. The top is a rearranged Pythagorean identity: $1-\cos^2\theta = \sin^2\theta$. 1−cos²=sin²
2. So the expression becomes $\dfrac{\sin^2\theta}{\sin\theta}$.
3. Cancel one factor of $\sin\theta$ from top and bottom: $\dfrac{\sin\theta\cdot\cancel{\sin\theta}}{\cancel{\sin\theta}}$. sin ÷ sin = 1

$\sin\theta$ **– three terms collapse to one.**



Example 9: dividing the Pythagorean identity by cos² turns the radius-1 triangle relation into 1 + tan² = 1/cos², true for every angle where cosine is non-zero.

Example 9

Prove the identity $1+\tan^2\theta = \dfrac{1}{\cos^2\theta}$.

SOLUTION

1. Start from the master identity $\sin^2\theta+\cos^2\theta=1$. the master
2. Divide every term by $\cos^2\theta$: $\dfrac{\sin^2\theta}{\cos^2\theta}+\dfrac{\cos^2\theta}{\cos^2\theta}=\dfrac{1}{\cos^2\theta}$. ÷ cos² throughout
3. The first term is $\left(\dfrac{\sin\theta}{\cos\theta}\right)^2 = \tan^2\theta$, and the middle term is 1.
4. Putting the pieces back gives $\tan^2\theta+1=\dfrac{1}{\cos^2\theta}$. rearranged

Proven: $1+\tan^2\theta=\dfrac{1}{\cos^2\theta}$ **– a second Pythagorean identity, born from dividing the first.**

Note: To *prove* an identity, transform one side by legal steps until it becomes the other. To *disprove* a claimed identity, you only need one angle where it fails – a single counterexample.

⚡ Mental-Math Shortcut

Treat $\sin^2\theta + \cos^2\theta = 1$ like the partners of a pair that always sums to 1. If you spot one squared ratio, the other is just "1 minus it". Know $\sin 30^\circ = \frac{1}{2}$? Then $\sin^2 30^\circ = \frac{1}{4}$, so $\cos^2 30^\circ = 1 - \frac{1}{4} = \frac{3}{4}$ instantly – and $\cos 30^\circ = \sqrt{\frac{3}{4}} = \frac{\sqrt{3}}{2}$. You never have to memorise both squares: one plus a subtraction gives the other.

🧠 The Logic of It

An identity makes a **universal** claim: $\cos^2\theta + \sin^2\theta = 1$ for *every* angle, all infinitely many of them. You cannot prove a "for all" statement by examples alone – checking 30°, 45° and 60° builds confidence, but it would never rule out some sneaky angle where it fails. (Recall Chapter 22: a single counterexample destroys a universal claim, but no finite list of confirmations *establishes* one.) So how do we know it truly holds for all θ?

We give a **general argument**, one that never names a specific angle. Take *any* angle θ whatsoever. By the unit-circle definition, the point $(\cos\theta, \sin\theta)$ lies on the circle of radius 1. Every point on that circle satisfies $x^2 + y^2 = 1$ – that is what "radius 1" *means*, by the distance formula. Substituting the coordinates gives $\cos^2\theta + \sin^2\theta = 1$. Because we assumed nothing about θ beyond its being an angle, the same chain of reasoning works for all of them at once. **That** is the difference between verifying and proving: verification tests instances; a proof argues about an *arbitrary* instance and so covers the whole infinite family in one stroke.

⚠ Common Mistakes

- Reading $\sin^2\theta$ as "sine of θ^2". It means $(\sin\theta)^2$ – compute the sine first, then square it.
- Forgetting the sign when square-rooting. $\cos^2\theta = \frac{16}{25}$ gives $\cos\theta = \pm\frac{4}{5}$; the quadrant decides which sign.
- Trying to prove an identity by plugging in a single angle. One value can *disprove* a false identity, but proving a true one needs a general argument.
- Writing $\sin\theta + \cos\theta = 1$. The identity has the ratios **squared**: $\sin^2\theta + \cos^2\theta = 1$.
- Using $\tan\theta = \dfrac{\sin\theta}{\cos\theta}$ where $\cos\theta = 0$ (e.g. $\theta = 90^\circ$). Division by zero makes the tangent undefined there.

Quiz

40.1.Q1. What is the Pythagorean identity?

A. $\sin^2\theta + \cos^2\theta = 1$

B. $\sin\theta + \cos\theta = 1$

C. $\tan^2\theta = 1$

D. $\sin\theta = \cos\theta$

40.1.Q2. An identity is an equation that is true for...

A. one special value

B. all allowed values

C. no values

D. only positive numbers

40.1.Q3. If $\sin\theta = \frac{3}{5}$ and θ is in Quadrant I, find $\cos\theta$.

40.1.Q4. Simplify $\sin^2\theta + \cos^2\theta$.

40.1.Q5. Rewrite $1 - \cos^2\theta$.

A. $\sin^2\theta$

B. $\tan^2\theta$

C. $\cos^2\theta$

D. 1

40.1.Q6. True or False: you can prove an identity holds for all angles by checking it at $30^\circ, 45^\circ, 60^\circ$.

A. True

B. False

40.1.Q7. What does $\tan\theta$ equal in terms of sine and cosine?

40.1.Q8. Expand and simplify $(1 - \cos\theta)(1 + \cos\theta)$.

🐍 From Concept to Code (Python)

An identity should pass *every* numerical test you throw at it. Let us check $\sin^2\theta + \cos^2\theta = 1$ at a spread of angles. Remember `math.sin` and `math.cos` take **radians**, so convert each degree first. The results will be 1 up to a tiny rounding wobble – proof in spirit, though only a real argument proves it for *all* angles.

```
import math

for deg in [0, 30, 45, 90, 200, 330]:
    t = math.radians(deg)
    total = math.sin(t)**2 + math.cos(t)**2
    print(deg, round(total, 10))
```

```
output
0 1.0
30 1.0
45 1.0
90 1.0
200 1.0
330 1.0
```

Try it:

1. Loop over `range(0, 360, 15)` and confirm the sum rounds to `1.0` at every angle.
2. Now *disprove* a fake identity: print `math.sin(t) + math.cos(t)` for the same angles and show it is **not** always 1 (try `0` and `90` degrees).

Answer Key – Chapter 40 – Trigonometric Identities

40.1.Q1. $\sin^2\theta + \cos^2\theta = 1$ – it holds for every angle.

40.1.Q2. All allowed values of the variable.

40.1.Q3. $\frac{4}{5}$ – $\cos^2\theta = 1 - \frac{9}{25} = \frac{16}{25}$, positive in Quadrant I.

40.1.Q4. 1 – that is the Pythagorean identity.

40.1.Q5. $\sin^2\theta$ – rearrange $\sin^2\theta + \cos^2\theta = 1$.

40.1.Q6. False – examples build confidence but a universal claim needs a general proof.

40.1.Q7. $\frac{\sin\theta}{\cos\theta}$ – valid wherever $\cos\theta \neq 0$.

40.1.Q8. $\sin^2\theta$ – it equals $1 - \cos^2\theta$, which is $\sin^2\theta$.

Chapter 41 · Inverse Trigonometric Functions

How to run trigonometry backwards: given a ratio of sides (or a coordinate), recover the angle that produced it, using arcsin, arccos and arctan – and how to pick the right quadrant.

The Idea Behind It

Why $\arcsin$ is only allowed to return angles in $[-90^\circ,\ 90^\circ]$, and cannot just "undo" sine on the whole circle. *but why?*

To **invert** a function you must be able to read its input back from its output. That only works if no two inputs share an output, the function must be **one-to-one** (Chapter 29: each input has exactly one output, so a true inverse needs each output to come from exactly one input). Sine fails this test, because it repeats:

$$\boxed{\sin 30^\circ = \tfrac{1}{2} \quad \text{and} \quad \sin 150^\circ = \tfrac{1}{2}}$$

Two *different* angles, one *same* sine. So the bare question "the angle whose sine is $\frac{1}{2}$" has more than one honest answer:

$$\sin^{-1}\left(\tfrac{1}{2}\right) = 30^\circ \ ? \quad \text{or} \quad 150^\circ \ ? \quad \text{or} \ 30^\circ + 360^\circ \ ? \ \ldots$$

A function is forbidden from returning two values for one input, so $\arcsin$ would not be a function at all. The fix is to **restrict the domain** first: keep only the slice -90° to 90°, where sine climbs steadily and never repeats a value, so it *is* one-to-one. On that single slice the inverse is well defined:

$$\therefore\ \arcsin\left(\tfrac{1}{2}\right) = 30^\circ \quad (\text{the one answer in } [-90^\circ,\ 90^\circ])$$

So an inverse never "undoes the whole circle", only its chosen branch, which is why a calculator's answer can still need a quadrant fix. *an inverse needs a one-to-one function*

Ordinary trigonometry takes an **angle** and gives back a **ratio**: type 30° into $\sin$ and out comes 0.5. But real problems usually run the other way. A ramp rises 3 metres over a 5-metre slope – *what angle* does it make with the ground? A robot arm must point at a target – *what angle* should the motor turn? To answer these we need to undo a trig function: feed in the ratio and recover the angle. That reverse operation is an **inverse trigonometric function**.

INVERSE TRIG FUNCTION

An **inverse trig function** takes a ratio and returns the angle that produces it. The three main ones are **arcsine** ($\arcsin$, also written $\sin^{-1}$), **arccosine** ($\arccos$, $\cos^{-1}$) and **arctangent** ($\arctan$, $\tan^{-1}$). Read $\arcsin(0.5)$ as "the angle whose sine is 0.5".

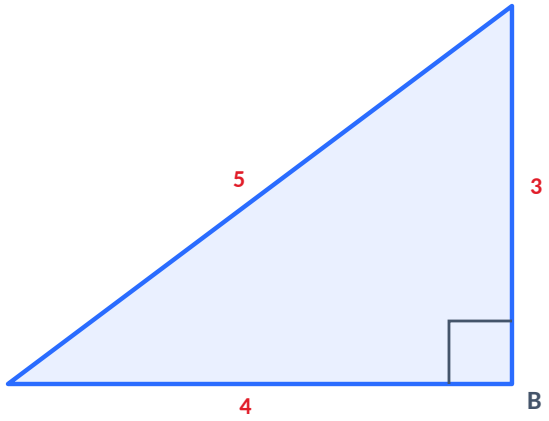
The little -1 in $\sin^{-1}$ is **not** a power. $\sin^{-1}(x)$ does **not** mean $1/\sin(x)$. It is the *inverse-function* mark, the same way that f^{-1} undoes f. To avoid this trap, this chapter prefers the names $\arcsin$, $\arccos$, $\arctan$.

Reading the angle off a right triangle

Recall **SOH-CAH-TOA**, the three right-triangle ratios:

$$\sin\theta = \frac{\text{opposite}}{\text{hypotenuse}}, \quad \cos\theta = \frac{\text{adjacent}}{\text{hypotenuse}}, \quad \tan\theta = \frac{\text{opposite}}{\text{adjacent}}.$$

Each inverse simply solves one of these for θ. If you know the *opposite* and the *hypotenuse*, use $\arcsin$. If you know *adjacent* and *hypotenuse*, use $\arccos$. If you know *opposite* and *adjacent*, use $\arctan$. The picture below is the ramp: opposite $= 3$, adjacent $= 4$, hypotenuse $= 5$.



A 3-4-5 right triangle. The angle θ sits at the bottom-left: opposite = 3, adjacent = 4, hypotenuse = 5. The square marks the right angle.

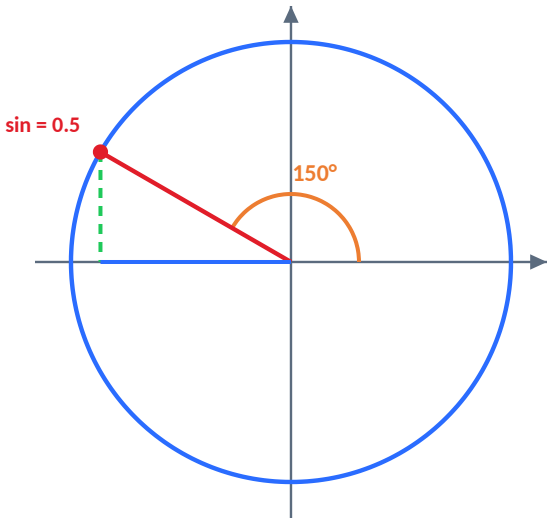
From this one triangle we can recover the angle θ three different ways, and they must all agree: $\arcsin\frac{3}{5} = \arccos\frac{4}{5} = \arctan\frac{3}{4} \approx 36.87^\circ$.

arcsine arccosine arctangent

Why each inverse has a limited range

There is a catch. Many different angles share the same sine: $\sin 30^\circ = 0.5$ but also $\sin 150^\circ = 0.5$, and you could keep going round the circle forever. If $\arcsin$ is to be a genuine **function** it must return *exactly one* answer (Chapter 29: one input, one output). So mathematicians agreed on a **principal range** for each inverse – a single slice of angles it is allowed to return.

- $\arcsin$ returns an angle in $[-90^\circ, 90^\circ]$ (right half of the circle).
- $\arccos$ returns an angle in $[0^\circ, 180^\circ]$ (top half of the circle).
- $\arctan$ returns an angle in $(-90^\circ, 90^\circ)$.



Both 30° and 150° have sine 0.5 (same height). arcsin must choose just one – it picks 30°, the angle inside −90° to 90°.

Note: Because $\arcsin$ only ever returns -90° to 90°, it can miss the angle you actually want. The ramp angle 36.87° is fine, but if a true angle lives in the second or third quadrant you must adjust the calculator answer by hand – see Example 5 and the $\arctan$ vs atan2 note.

Worked examples

Example 1

A ramp rises 3 m over a slope (hypotenuse) of 5 m. What angle does it make with the ground?

SOLUTION

1. We know the **opposite** (3) and the **hypotenuse** (5), so use $\arcsin$. *opp & hyp → arcsin*
2. Form the ratio: $\sin\theta = \dfrac{\text{opp}}{\text{hyp}} = \dfrac{3}{5} = 0.6$.
3. Take the inverse: $\theta = \arcsin(0.6)$. *undo the sine*
4. Evaluate (calculator in degree mode): $\arcsin(0.6) \approx 36.87^\circ$.

The ramp rises at about 36.87°.

Example 2

Find the angle whose cosine is 0.5.

SOLUTION

1. This is asking for $\arccos(\ 0.5\ )$ directly.
2. Recall the special triangles: $\cos\ 60^\circ\ =\ 0.5$. *which angle gives 0.5?*
3. Check 60° lies in the $\arccos$ range $[0^\circ, 180^\circ]$ – it does. *in range ✓*

$\arccos(0.5) = 60^\circ$.

Example 3

A right triangle has opposite $= 4$ and adjacent $= 4$. Find the angle.

SOLUTION

1. Opposite and adjacent are known, so use $\arctan$. *opp & adj → arctan*
2. $\tan\theta = \dfrac{\text{opp}}{\text{adj}} = \dfrac{4}{4} = 1$.
3. $\theta = \arctan(\ 1\ )$. *undo the tangent*
4. The angle with tangent 1 is the one where opposite equals adjacent: 45°.

$\theta = 45^\circ$. **(A right triangle with two equal legs is always 45°-45°-90°.)**

Example 4

Recover the angle for the 3-4-5 triangle three different ways and check they match.

SOLUTION

1. Using sides opposite/hyp: $\arcsin\ \frac{3}{5} = \arcsin(0.6) \approx 36.87^\circ$. *SOH backwards*
2. Using adjacent/hyp: $\arccos\ \frac{4}{5} = \arccos(0.8) \approx 36.87^\circ$. *CAH backwards*
3. Using opposite/adjacent: $\arctan\ \frac{3}{4} = \arctan(0.75) \approx 36.87^\circ$. *TOA backwards*

All three agree: $\theta \approx 36.87^\circ$. The same angle, found from whichever pair of sides you happen to know.

Example 5

A point sits at $(-4, 3)$. Find the angle its direction makes, measured anticlockwise from the positive x-axis.

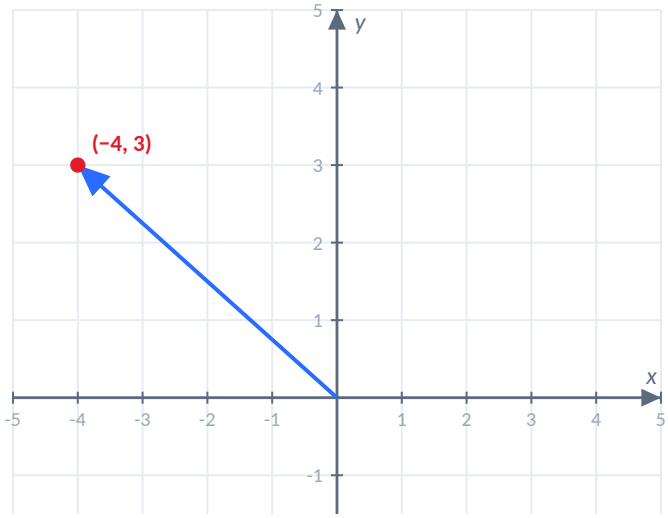
SOLUTION

1. Naively, $\tan\theta = \dfrac{y}{x} = \dfrac{3}{-4} = -0.75$, so a calculator gives $\arctan(-0.75) \approx -36.87^\circ$.

   wrong quadrant!

2. But $(-4, 3)$ is in the **second quadrant** (left and up), where the true angle is between 90° and 180° – not -36.87°. *−x, +y → Q2*
3. $\arctan$ only returns -90° to 90°, so add 180° to land in the correct quadrant: $-36.87^\circ + 180^\circ = 143.13^\circ$. *+180° fix*
4. A tool that does this for you automatically is $\operatorname{atan2}(y, x)$ – see the note below.

The direction is about 143.13°, not -36.87°. Quadrant always matters when recovering an angle from coordinates.



The point (−4, 3) lies up-and-left (second quadrant), so its direction angle 143.13° opens beyond 90°. A bare arctan would report −36.87°, pointing the wrong way entirely.

Example 6

You stand 10 m from the base of a tree and your eye must look up 6 m to see the top. What is the angle of elevation?

SOLUTION

1. Opposite (height above eye) $= 6$, adjacent (ground distance) $= 10$, so use $\arctan$. *opp & adj → arctan*
2. $\tan\theta = \dfrac{6}{10} = 0.6$.
3. $\theta = \arctan(0.6) \approx 30.96^\circ$. *undo the tangent*

You look up at about 31°.

Example 7

A child's slide is 4 m long (the hypotenuse) and drops 2 m in height. At what angle does it tilt from the horizontal?

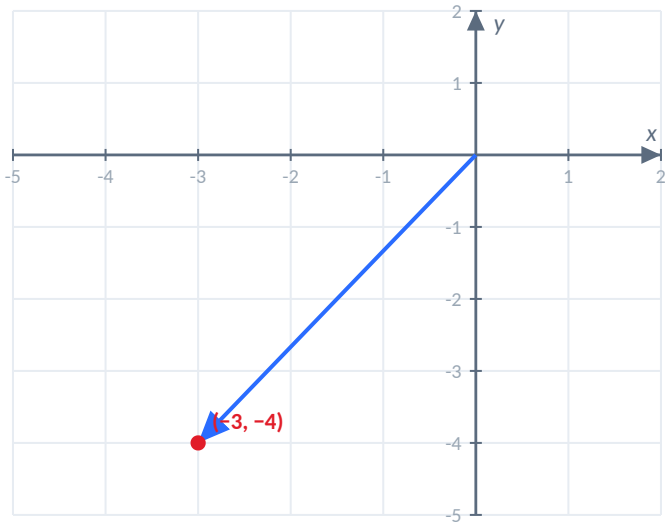
SOLUTION

1. The drop (2) is opposite the tilt angle and the slide (4) is the hypotenuse, so use $\arcsin$.

opp & hyp → arcsin

2. Form the ratio: $\sin\theta = \dfrac{\text{opp}}{\text{hyp}} = \dfrac{2}{4} = 0.5$.
3. Take the inverse: $\theta = \arcsin(0.5)$. *undo the sine*
4. The angle whose sine is exactly 0.5 is the special value 30°. *known signpost*

The slide tilts at 30°.



Example 8: the point (−3, −4) lies down-and-left in the third quadrant. Its true direction is about 233°, but a bare arctan would report only 53°, half a turn out.

Example 8

A point sits at $(-3, -4)$. Find the direction angle, measured anticlockwise from the positive x-axis (in the range 0° to 360°).

SOLUTION

1. Naively, $\tan\theta = \frac{y}{x} = \frac{-4}{-3} = +\frac{4}{3}$, so a calculator returns $\arctan(\frac{4}{3}) \approx 53.13^\circ$. looks Q1 – wrong!
2. But $(-3, -4)$ is in the **third quadrant** (left and down), where the true angle lies between 180° and 270°.
   -x, -y → Q3
3. Add 180° to the principal value to swing into Quadrant III: $53.13^\circ + 180^\circ = 233.13^\circ$. +180° fix

The direction is about 233.13°, not 53.13° – the bare arctan landed exactly half a turn away.

Example 9

A 6 m ladder leans against a wall with its foot 3 m out from the base. What angle does it make with the **ground**, and is it safe (the recommended range is 70° to 80°)?

SOLUTION

1. The foot distance (3, along the ground) is **adjacent** to the ground angle and the ladder (6) is the hypotenuse, so use $\arccos$. adj & hyp → arccos
2. Form the ratio: $\cos\theta = \frac{\text{adj}}{\text{hyp}} = \frac{3}{6} = 0.5$.
3. Take the inverse: $\theta = \arccos(0.5) = 60^\circ$. known signpost
4. Compare with the safe band 70°–80°: 60° falls below it. too shallow

The ladder sits at 60° – outside the safe 70°–80° range, so it is leaning too shallow.

Note: Calculators and code can work in **degrees** or **radians**. The inverse functions in Python (asin, acos, atan, atan2) return **radians**; convert with $\text{degrees}()$ if you want a human-friendly angle.

Mental-Math Shortcut

Memorise just **three signpost values** and you can sanity-check any inverse in your head: - $\arcsin(0) = 0^\circ$, $\arcsin(0.5) = 30^\circ$, $\arcsin(1) = 90^\circ$. - $\arccos(1) = 0^\circ$, $\arccos(0.5) = 60^\circ$, $\arccos(0) = 90^\circ$. - $\arctan(0) = 0^\circ$, $\arctan(1) = 45^\circ$.

Notice the mirror: as a sine grows from 0 to 1 the angle climbs $0^\circ \to 90^\circ$, but as a cosine grows from 0 to 1 the angle *falls* $90^\circ \to 0^\circ$. So $\arcsin$ and $\arccos$ of the same input add up to 90°: $\arcsin(0.6) + \arccos(0.6) = 90^\circ$. If a calculator answer is wildly off one of these signposts, you have probably left it in radian mode.

🧠 The Logic of It

An inverse function only makes sense when the original function is **one-to-one** – each output coming from exactly one input. Sine fails this test on its own: $\sin 30^\circ$ and $\sin 150^\circ$ are *both* 0.5, so "the angle whose sine is 0.5" has two honest answers, and a function is forbidden from returning two outputs for one input (Chapter 29).

So we reason as follows. *If* we let $\arcsin$ accept the whole circle, *then* it is not a function – contradiction. Therefore we must **restrict the domain** of sine first: keep only the angles from -90° to 90°. On that slice, sine climbs steadily from -1 up to 1 and never repeats a value, so it *is* one-to-one, and its inverse is well defined. That single agreed slice is the **principal range**.

The consequence is a rule you must always apply: a calculator's $\arcsin$ / $\arccos$ / $\arctan$ gives the *principal* angle, which is correct **only if** your real angle lies in that range. When it does not – as with the second-quadrant point $(-4, 3)$ – the logic forces a correction (add 180°, or use $\operatorname{atan2}$). "Inverse exists" therefore really means "inverse exists *on a chosen slice*", and respecting that slice is the whole discipline of using these functions correctly.

⚠ Common Mistakes

- Reading $\sin^{-1}(x)$ as $\dfrac{1}{\sin x}$. The -1 marks the **inverse function**, not a reciprocal. The reciprocal of $\sin$ is $\csc$, a different thing.
- Leaving the calculator in **radian** mode and reporting, say, 0.6435 as "the angle in degrees". Match the mode to the units you want.
- Trusting $\arctan(y/x)$ for a direction without checking the **quadrant**. For points with negative x you must add 180° (or use $\operatorname{atan2}$).
- Feeding $\arcsin$ or $\arccos$ a number outside $[-1, 1]$. No angle has a sine of 1.4, so the function is undefined there (a real calculator errors).
- Forgetting the range limits: expecting $\arccos$ to return a negative angle. It never does – its output is always between 0° and 180°.

Quiz

41.1.Q1. What does $\arcsin(x)$ mean?

A. $1 \div \sin x$

B. The angle whose sine is x

C. The sine of angle x

D. $\sin x$ squared

41.1.Q2. Evaluate $\arctan(1)$ in degrees.

A. 30°

B. 45°

C. 60°

D. 90°

41.1.Q3. In a right triangle you know the **opposite** and the **hypotenuse**. Which inverse finds the angle?

A. $\arcsin$

B. $\arccos$

C. $\arctan$

D. none

41.1.Q4. True or False: $\sin^{-1}(x)$ is the same as $\dfrac{1}{\sin x}$.

A. True

B. False

41.1.Q5. A ramp has opposite $= 5$ and hypotenuse $= 10$. Find its angle.

41.1.Q6. What is the principal output range of $\arccos$?

A. $[-90^\circ, 90^\circ]$

B. $[0^\circ, 180^\circ]$

C. $[0^\circ, 90^\circ]$

D. all angles

41.1.Q7. A point is at $(-3, 4)$ (second quadrant). A calculator gives $\arctan(4/-3) \approx -53.13^\circ$. What is the true direction angle?

41.1.Q8. True or False: $\arcsin(2)$ is a valid angle.

A. True

B. False

41.1.Q9. Find $\arccos(1)$.

🐍 From Concept to Code (Python)

Python's `math` module runs trigonometry backwards with `asin`, `acos`, `atan` and the quadrant-aware `atan2`. Each returns **radians**, so we wrap them in `math.degrees(...)` to read a human angle. Here we recover the angle of the classic 3-4-5 triangle three ways and confirm they agree.

```
import math

opp, adj, hyp = 3, 4, 5
print(round(math.degrees(math.asin(opp / hyp)), 2))   # from opp & hyp
print(round(math.degrees(math.acos(adj / hyp)), 2))   # from adj & hyp
print(round(math.degrees(math.atan(opp / adj)), 2))   # from opp & adj
```

```
output
36.87
36.87
36.87
```

```
import math

# atan2(y, x) handles the quadrant automatically.
# The point (-4, 3) is in the second quadrant.
angle = math.degrees(math.atan2(3, -4))
print(round(angle, 2))

# Plain atan would wrongly give -36.87; atan2 fixes it:
print(round(math.degrees(math.atan(3 / -4)), 2))
```

```
output
143.13
-36.87
```

Try it:

1. Find the angle of elevation to a tree top that is 6 m above your eye when you stand 10 m away: print `math.degrees(math.atan(6/10))`.
2. Try `math.atan2(y, x)` for the points `(1, 1)`, `(-1, 1)`, `(-1, -1)` and `(1, -1)`. Confirm you get 45, 135, −135 and −45 degrees.

Answer Key – Chapter 41 – Inverse Trigonometric Functions

41.1.Q1. The angle whose sine is x – it runs sine backwards.

41.1.Q2. 45° – the angle whose tangent is 1 (equal opposite and adjacent).

41.1.Q3. $\arcsin$ – because $\sin\theta = \text{opp}/\text{hyp}$.

41.1.Q4. False – the -1 denotes the inverse function, not a reciprocal.

41.1.Q5. 30° – $\sin\theta = 5/10 = 0.5$, and $\arcsin(0.5) = 30^\circ$.

41.1.Q6. $[0^\circ, 180^\circ]$ – arccos never returns a negative angle.

41.1.Q7. 126.87° – add 180° because the point is in the second quadrant.

41.1.Q8. False – no angle has sine 2; the input must lie in $[-1, 1]$.

41.1.Q9. 0° – the only angle (in range) with cosine 1 is 0°.

Chapter 42 · Waves & Periodic Functions

How to read the sine and cosine graphs as repeating waves, and how the four numbers in $y = A\sin(Bx + C)$ stretch, squeeze and slide that wave – its amplitude, period and phase.

The Idea Behind It

Why $y = \sin(Bx)$ repeats every $\frac{2\pi}{B}$, not every 2π. *but why?*

The value $\sin(\text{angle})$ is the height of a point on the unit circle, and that height returns to where it started only after a **full revolution**, an inside angle of 2π. So one period is "however far x must travel to push the inside angle through one whole lap":

$$\boxed{\text{one period} \iff \text{inside angle increases by } 2\pi}$$

For plain $\sin x$ the inside angle *is* x, so a lap needs $x = 2\pi$. But inside $\sin(Bx)$ the angle is Bx, which grows B times as fast. To complete one lap we set the inside angle equal to 2π and solve for x:

$$Bx = 2\pi \implies x = \frac{2\pi}{B} = \frac{2\pi}{B}$$

A bigger B winds the angle round faster, so the wave finishes sooner, a *shorter* period. The conclusion drops straight out:

$$\therefore \text{ period of } \sin(Bx) = \frac{2\pi}{B}$$

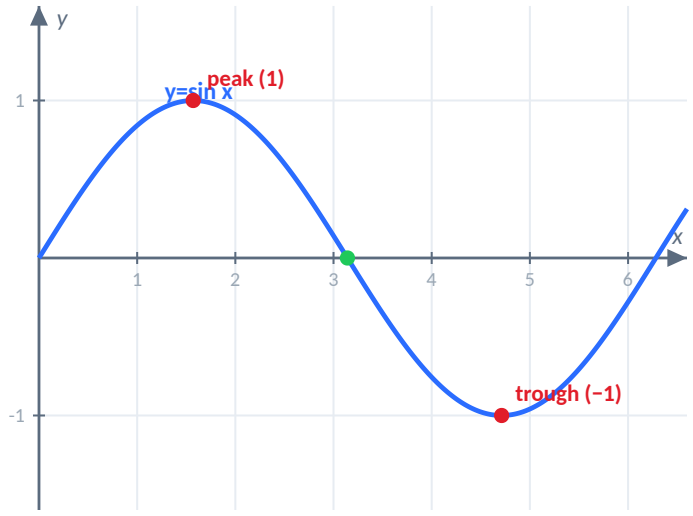
The period is set by one idea, the angle completing a single revolution, so no graphing or measuring is needed to find it. *one period = one full turn*

Sound, light, radio, the tides, a swinging pendulum, alternating current in a wall socket – all of them rise and fall in the same smooth, repeating shape. That shape is a **wave**, and the purest example is the graph of the sine function. Once you can read one wave, you can read them all, because every wave is just a sine that has been stretched taller, squeezed narrower, or slid sideways.

PERIODIC FUNCTION

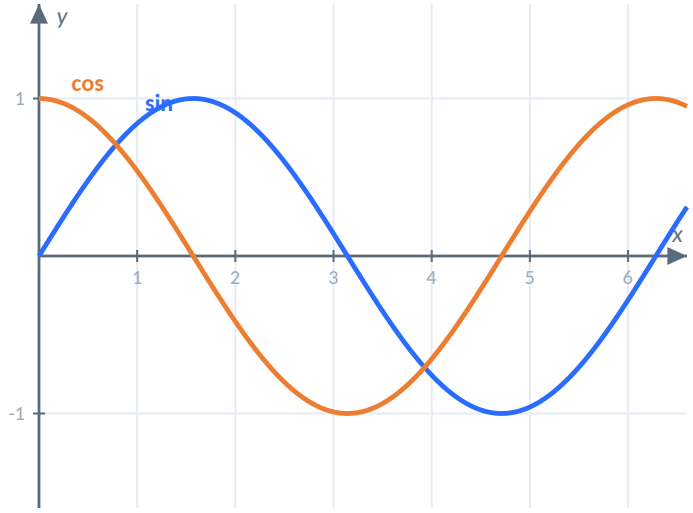
A function is **periodic** if its graph repeats the same pattern over and over at a fixed spacing. The length of one full repeat is the **period**. Sine and cosine are periodic with period 2π (one full trip around the unit circle).

Picture the unit circle from earlier chapters. As an angle x sweeps round, the **height** of the point is $\sin x$. Plot that height against the angle and the dot traces a wave: up to 1, back through 0, down to -1, and back – then it repeats forever.



One period of y = sin x, from 0 to 2π (≈6.28). It peaks at +1 (a quarter turn), crosses 0 at π, bottoms at −1, and returns.

Cosine is the very same wave, just **shifted**: it starts at its peak instead of at zero. In fact $\cos x = \sin(x + 90^\circ)$ – cosine is sine that left the gate a quarter-period early.



sin (blue) starts at 0 and rises; cos (orange) starts at 1. They are the same wave shifted by a quarter period.

The three knobs of a wave

A general wave is written

$$y = A\ \sin(Bx + C)$$

and each letter is a "knob" that changes the wave in one specific way:

AMPLITUDE

The **amplitude** is $|A|$: how far the wave reaches above and below the centre line. $A = 2$ makes a wave twice as tall (peaks at $+2$, troughs at -2); $A = \frac{1}{2}$ makes it half as tall. It controls *height*, never width.

PERIOD

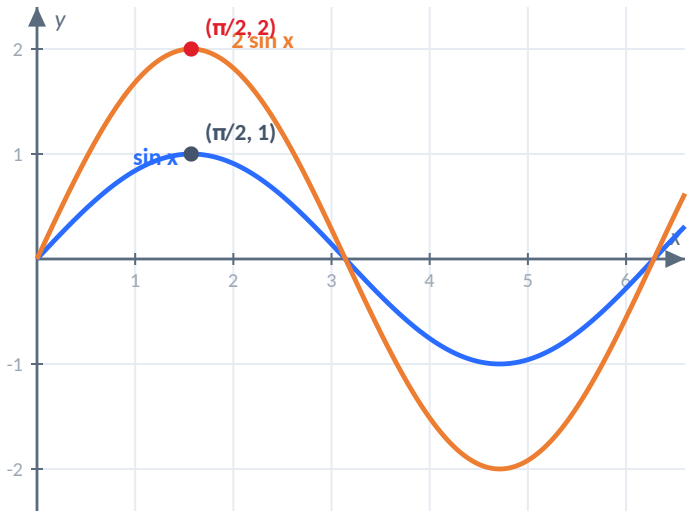
The **period** is how long one full repeat takes. For $\sin(Bx)$ it equals $\dfrac{2\pi}{|B|}$. A larger B packs more waves into the same space (shorter period); $B = 2$ doubles the frequency and halves the period. The **frequency** is how many repeats fit in 2π, namely $|B|$.

PHASE SHIFT

The **phase shift** slides the whole wave left or right. From $A\sin(Bx + C)$ the shift is $-\dfrac{C}{B}$: a positive C moves the wave **left**. It changes *where* the wave starts, not its height or width.

wave amplitude period phase shift

Below, the plain wave $y = \sin x$ is drawn against $y = 2\sin x$. Same crossings, same period – but the orange wave is stretched to twice the height. Amplitude changes height alone.



Amplitude. y = 2 sin x (orange) peaks at 2; y = sin x (blue) peaks at 1. The shape and zeros are identical – only the height doubles.

Worked examples

Example 1

State the amplitude and period of $y = 3\sin x$.

SOLUTION

1. Compare with $y = A\sin(Bx + C)$: here $A = 3$, $B = 1$, $C = 0$. match the form
2. Amplitude $= |A| = 3$. A is the height
3. Period $= \dfrac{2\pi}{|B|} = \dfrac{2\pi}{1} = 2\pi$.

Amplitude 3 (peaks at $+3$, troughs at -3); period 2π (unchanged width).

Example 2

Find the amplitude and period of $y = \sin(2x)$.

SOLUTION

1. Here $A = 1$, $B = 2$, $C = 0$. no number out front → A=1
2. Amplitude $= |A| = 1$ – same height as the plain wave.
3. Period $= \dfrac{2\pi}{|B|} = \dfrac{2\pi}{2} = \pi$. bigger B → shorter

Amplitude 1, period π – the wave repeats twice as often (two full waves in the usual space of one).

Example 3

A wave is $y = 4\sin(2x)$. Give its amplitude, period and maximum value.

SOLUTION

1. Read off $A = 4$, $B = 2$.
2. Amplitude $= |A| = 4$, so the maximum value is $+4$ and the minimum is -4. max = A
3. Period $= \dfrac{2\pi}{2} = \pi$.

Amplitude 4, period π, maximum $+4$.

Example 4

Where does $y = \sin(x - \frac{\pi}{2})$ start, compared with $y = \sin x$?

SOLUTION

1. Write it as $\sin(Bx + C)$ with $B = 1$ and $C = -\frac{\pi}{2}$.
2. Phase shift $= -\frac{C}{B} = -\frac{-\pi/2}{1} = +\frac{\pi}{2}$. shift = -C/B
3. A positive shift moves the wave to the **right** by $\frac{\pi}{2}$. + shift → right

It is $\sin x$ slid right by $\frac{\pi}{2}$ – which is exactly $-\cos x$... and indeed $\cos x = \sin(x + \frac{\pi}{2})$.

Example 5

A mass on a spring has height $y = 5\sin(2t)$ centimetres after t seconds. How tall is the swing, and how long is one bounce?

SOLUTION

1. Amplitude $= 5$, so the mass moves 5 cm above and 5 cm below centre – a total swing of 10 cm.
   peak-to-trough = 2A
2. Period $= \frac{2\pi}{B} = \frac{2\pi}{2} = \pi \approx 3.14$ seconds. B=2

The swing is 10 cm peak-to-peak, and one complete bounce takes about 3.14 seconds.

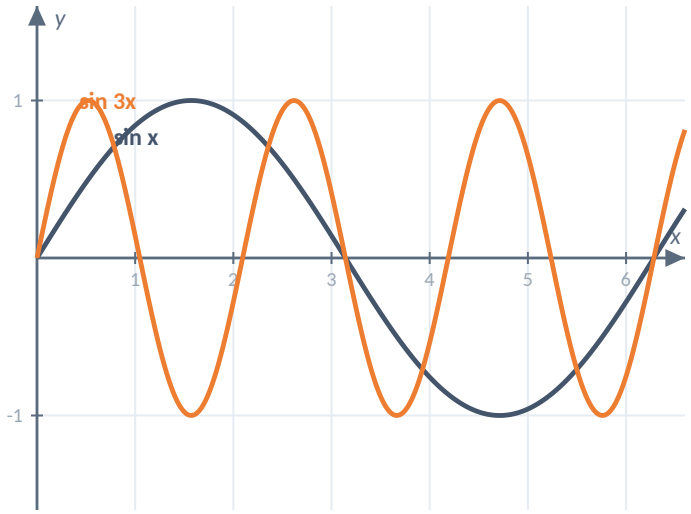
Example 6

Build $y = A\sin(Bx + C)$ with amplitude 2, period π and no phase shift.

SOLUTION

1. Amplitude 2 means $A = 2$. A sets height
2. Period π means $\frac{2\pi}{B} = \pi$, so $B = 2$. solve for B
3. No phase shift means $C = 0$.

$y = 2\sin(2x)$.



Example 7: y = sin 3x (orange) fits three full waves into the span 0 to 2π, so its period is one-third of the plain wave (grey). Same height, triple the frequency.

Example 7

Find the amplitude, period and frequency of $y = \sin(3x)$.

SOLUTION

1. Match $y = A\sin(Bx + C)$: here $A = 1$, $B = 3$, $C = 0$. no front number → A=1
2. Amplitude $= |A| = 1$ – the same height as the plain wave.
3. Period $= \dfrac{2\pi}{|B|} = \dfrac{2\pi}{3} = \dfrac{2\pi}{3}$. bigger B → shorter
4. Frequency $= |B| = 3$ – three full waves fit between 0 and 2π.

Amplitude 1, period $\frac{2\pi}{3}$, frequency 3.

Example 8

For $y = 3\sin(2x + \pi)$, give the amplitude, period and phase shift.

SOLUTION

1. Read off $A = 3$, $B = 2$, $C = \pi$. match the form
2. Amplitude $= |A| = 3$; period $= \dfrac{2\pi}{|B|} = \dfrac{2\pi}{2} = \pi$.
3. Phase shift $= -\dfrac{C}{B} = -\dfrac{\pi}{2} = -\dfrac{\pi}{2}$. shift = -C/B
4. A negative shift slides the wave to the **left** by $\frac{\pi}{2}$. +C → left here

Amplitude 3, period π, phase shift $-\frac{\pi}{2}$ (left by $\frac{\pi}{2}$).

Example 9

A Ferris wheel rider's height swings between 2 m and 40 m above the ground, completing one loop every 60 seconds. Write a cosine model $y = A\cos(Bt) + D$ for the height.

SOLUTION

1. The centre line D is the average of top and bottom: $D = \dfrac{40+2}{2} = 21$ m. *midline = average*
2. The amplitude is half the full swing: $A = \dfrac{40-2}{2} = 19$ m. *half the range*
3. One loop every 60 s means period 60, so $\dfrac{2\pi}{B} = 60$, giving $B = \dfrac{2\pi}{60} = \dfrac{\pi}{30}$. *solve for B*

$y = 19\cos\left(\frac{\pi}{30}\,t\right) + 21$ **– starting at the top (40 m) when $t = 0$.**

Note: "Peak-to-trough" height is $2A$, not A. The amplitude A is measured from the **centre line** to a peak, so a wave with amplitude 5 swings a total of 10 from its lowest point to its highest.

⚡ Mental-Math Shortcut

Read a wave $y = A\sin(Bx + C)$ off in three quick glances, no graphing needed: - **Tall or short?** Look at A. The number out front *is* the amplitude. Peaks at $+A$, troughs at $-A$. - **Fast or slow?** Look at B. Bigger B = more waves squeezed in = shorter period $2\pi/B$. A handy rule: B **tells you how many full waves fit between 0 and 2π.** So $\sin(3x)$ fits three. - **Started early or late?** Look at the sign of C. **Plus shifts left, minus shifts right** (the opposite of what the sign looks like), by C/B.

So $y = 5\sin(2x)$ is "height 5, two waves per cycle, no slide" before you draw a single point.

🧠 The Logic of It

Why must a sine wave repeat *exactly* every 2π, forever, and never drift? Trace the definition. The value $\sin x$ is the height of a point that has travelled an angle x around the unit circle. Going 2π further means going **one whole lap** and returning to the very same point on the circle. The same point has the same height. Therefore

$$\sin(x + 2\pi) = \sin x \quad \text{for every } x.$$

That "for every x" is the key: it is a *universal* statement (Chapter 22's $\forall$), not a claim about one lucky value. Because it holds at **every** input, the entire graph from 2π onward is a perfect copy of the graph before it, and the copy after that, and so on without end. Periodicity is not an accident of the picture; it is forced, point by point, by the geometry of returning to where you started.

The same reasoning fixes the period of a squeezed wave. In $\sin(Bx)$, the inside angle increases B times as fast, so it completes a full lap (2π of *inside* angle) after only $x = 2\pi/B$ of *outside* distance. Set "inside angle $= 2\pi$", solve, and the period $2\pi/B$ drops out by pure logic – no measuring required.

⚠ Common Mistakes

- Thinking B changes the height. B controls the **period** (width), not the amplitude. Only A changes how tall the wave is.
- Getting the phase-shift direction backwards. In $\sin(Bx + C)$ a **positive** C slides the wave **left**, and you divide by B: the shift is $-C/B$, not C.
- Forgetting to divide by B when finding the period: the period of $\sin(2x)$ is $\frac{2\pi}{2} = \pi$, not 2π.
- Reporting the full peak-to-trough distance as the amplitude. Amplitude is from the **centre** to a peak (A); the full swing is $2A$.
- Confusing period and frequency. The **period** is the length of one repeat; the **frequency** is how many repeats fit in a given span. They are reciprocals of each other (up to the 2π).

Quiz

42.1.Q1. What is the amplitude of $y = 4\sin x$?

A. 1

B. 2

C. 4

D. 8

42.1.Q2. What is the period of $y = \sin x$?

A. π

B. 2π

C. $\frac{\pi}{2}$

D. 1

42.1.Q3. Find the period of $y = \sin(2x)$.

A. 2π

B. 4π

C. π

D. $\frac{\pi}{2}$

42.1.Q4. True or False: changing A in $y = A\sin x$ changes the period.

A. True

B. False

42.1.Q5. For $y = 3\sin(2x)$, what is the maximum value the wave reaches?

42.1.Q6. How are $\sin x$ and $\cos x$ related as waves?

A. Different shapes

B. Cosine is sine shifted by a quarter period

C. Cosine is upside-down sine

D. They are identical

42.1.Q7. A wave has amplitude 6. What is its total peak-to-trough height?

42.1.Q8. Write a sine wave with amplitude 2 and period π (no phase shift).

🐍 From Concept to Code (Python)

Computers draw a wave by **sampling** it: pick many evenly spaced inputs, compute $\sin$ at each, and you have a list of heights ready to plot. Using only `math`, we sample one period of $y = \sin x$ in 8 steps, then build a general wave $y = A\sin(Bx + C)$ and read its amplitude straight from the samples.

```
import math

# Sample one period of sin x in 8 even steps (0 up to 7pi/4)
for i in range(8):
    x = 2 * math.pi * i / 8
    print(f"x={x:.2f}  sin(x)={round(math.sin(x), 2)}")
```

```
output
x=0.00  sin(x)=0.0
x=0.79  sin(x)=0.71
x=1.57  sin(x)=1.0
x=2.36  sin(x)=0.71
x=3.14  sin(x)=0.0
x=3.93  sin(x)=-0.71
x=4.71  sin(x)=-1.0
x=5.50  sin(x)=-0.71
```

```
import math

def wave(x, A=2.0, B=1.0, C=0.0):   # y = A sin(Bx + C)
    return A * math.sin(B * x + C)

values = [round(wave(2 * math.pi * i / 8), 2) for i in range(8)]
print(values)
print("amplitude =", max(abs(v) for v in values))
```

```
output
[0.0, 1.41, 2.0, 1.41, 0.0, -1.41, -2.0, -1.41]
amplitude = 2.0
```

Try it:

1. Change `A` to `3` in `wave` and confirm the amplitude printed becomes `3.0`.
2. Set `B=2` and sample `range(16)` over `[0, 2pi)`. Count how many times the value returns to near `0` – you should see two full waves.

Answer Key – Chapter 42 – Waves & Periodic Functions

42.1.Q1. 4 – the number in front of $\sin$ is the amplitude; peaks reach $+4$.

42.1.Q2. 2π – one full trip around the unit circle.

42.1.Q3. π – the period is $2\pi/B = 2\pi/2 = \pi$.

42.1.Q4. False – A changes the height (amplitude); the period depends only on B.

42.1.Q5. 3 – the maximum equals the amplitude $|A| = 3$.

42.1.Q6. Cosine is sine shifted left by a quarter period: $\cos x = \sin(x + \frac{\pi}{2})$.

42.1.Q7. 12 – the full swing is $2A = 2 \times 6 = 12$.

42.1.Q8. $y = 2\sin(2x)$ – $A = 2$ gives the amplitude, and $B = 2$ gives period $2\pi/2 = \pi$.

Chapter 43 · Rotations & Polar Coordinates

How to describe a point by its distance and direction instead of its x and y, convert freely between the two, and rotate any point about the origin using the rotation formulas.

The Idea Behind It

Why rotating (x, y) by θ gives $(x\cos\theta - y\sin\theta,\ x\sin\theta + y\cos\theta)$, signs and all. *but why?*

The formulas look like an arbitrary recipe, but they are just "spin the angle, keep the distance" written in x- and-y language. Write the point in **polar** form, distance r and direction φ. By the conversion rules $x = r\cos\varphi$ and $y = r\sin\varphi$. A rotation does exactly one thing, add θ to the angle and leave r alone, so the rotated point is:

$$x' = r\cos(\ \varphi + \theta\ ), \qquad y' = r\sin(\ \varphi + \theta\ )$$

Now expand each with the **angle-addition formulas**, the rule the whole proof leans on:

$$\boxed{\begin{aligned}\cos(\varphi+\theta) &= \cos\varphi\cos\theta - \sin\varphi\sin\theta\\ \sin(\varphi+\theta) &= \sin\varphi\cos\theta + \cos\varphi\sin\theta\end{aligned}}$$

Multiply through by r and regroup, then substitute back $r\cos\varphi = x$ and $r\sin\varphi = y$:

$$x' = (\ r\cos\varphi\ )\cos\theta - (\ r\sin\varphi\ )\sin\theta = x\cos\theta - y\sin\theta$$

$$y' = (\ r\sin\varphi\ )\cos\theta + (\ r\cos\varphi\ )\sin\theta = x\sin\theta + y\cos\theta$$

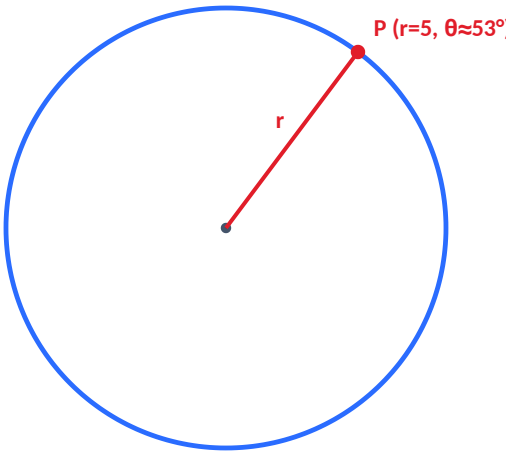
The minus on x' and the plus on y' are not chosen by hand: they are exactly the signs the cosine and sine addition rules carry.

rotation = add the angle, kept honest by angle-addition

There are two natural ways to say *where* a point is. The first, which you already know, is **Cartesian**: go x across and y up, like a street grid. The second is **polar**: how *far* away is it, and in *what direction*? "Three blocks east and four north" versus "five blocks, roughly north-east" describe the same corner. Polar coordinates are the language of anything that turns or radiates – radar, clock hands, spinning game sprites, a planet's orbit – so being able to switch between the two is a core skill.

POLAR COORDINATES

In **polar coordinates** a point is written (r, θ), where r is its **distance from the origin** and θ is the **angle** its direction makes, measured anticlockwise from the positive x-axis. The origin itself is the **pole**; the positive x-axis is the **polar axis**.



A polar point P: the red arm is the distance r from the pole (centre); the angle θ is measured anticlockwise from the right. Here r = 5 and θ ≈ 53°, which is the Cartesian point (3, 4).

Converting polar to Cartesian

Drop a right triangle from the point to the x-axis. The arm r is the hypotenuse, the horizontal leg is x, the vertical leg is y. Straight from SOH-CAH-TOA:

$$x = r\cos\theta, \qquad y = r\sin\theta.$$

So polar $\rightarrow$ Cartesian is just "multiply the distance by cosine for x, by sine for y".

Converting Cartesian to polar

Going back, the distance comes from Pythagoras and the angle from arctangent (Chapter 41):

$$r = \sqrt{x^2 + y^2}, \qquad \theta = \operatorname{atan2}(y, x).$$

We use $\operatorname{atan2}(y, x)$ rather than plain $\arctan(y/x)$ because it inspects the signs of x and y and returns the angle in the **correct quadrant** automatically – exactly the quadrant trap from the inverse-trig chapter.

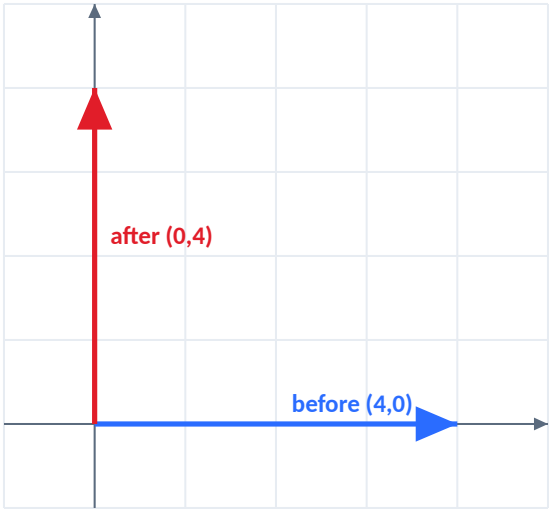
polar coordinates pole rotation

Rotating a point about the origin

A **rotation** turns a point around the origin by an angle α, keeping its distance r the same and adding α to its direction. You *could* convert to polar, add α, and convert back – but the algebra collapses to two tidy formulas. To rotate (x, y) anticlockwise by α:

$$x' = x\cos\alpha - y\sin\alpha,$$
$$y' = x\sin\alpha + y\cos\alpha.$$

The picture shows the simplest case: the point $(4, 0)$ rotated 90° swings up to $(0, 4)$, same distance from the origin, quarter-turn anticlockwise.



Rotating (4, 0) by 90° anticlockwise gives (0, 4). The arrow keeps its length 4 and turns a quarter circle.

Worked examples

Example 1

Convert the polar point $(r, \theta) = (5, 53.13^\circ)$ to Cartesian coordinates.

SOLUTION

1. Use $x = r\cos\theta$: $x = 5\cos 53.13^\circ \approx 5(0.6) = 3$. cos → x
2. Use $y = r\sin\theta$: $y = 5\sin 53.13^\circ \approx 5(0.8) = 4$. sin → y

The Cartesian point is $(3, 4)$ – the familiar 3-4-5 triangle, seen as a distance and a direction.

Example 2

Convert the Cartesian point $(3, 4)$ back to polar.

SOLUTION

1. Distance: $r = \sqrt{3^2 + 4^2} = \sqrt{9 + 16} = \sqrt{25} = 5$. Pythagoras
2. Angle: $\theta = \operatorname{atan2}(4, 3) = \arctan\frac{4}{3} \approx 53.13^\circ$ (first quadrant, so plain arctan is fine here).

The polar form is $(5,\ 53.13^\circ)$ – matching Example 1, as it must.

Example 3

Convert the polar point $(2, 60^\circ)$ to Cartesian coordinates.

SOLUTION

1. $x = 2\cos 60^\circ = 2(0.5) = 1$. cos → x
2. $y = 2\sin 60^\circ = 2(0.866) \approx 1.73$. sin → y

The Cartesian point is approximately $(1,\ 1.73)$.

Example 4

Rotate the point $(4, 0)$ anticlockwise by 90°.

SOLUTION

1. At 90°: $\cos 90^\circ = 0$ and $\sin 90^\circ = 1$. quarter-turn values
2. $x' = x\cos\alpha - y\sin\alpha = 4(0) - 0(1) = 0$. minus on x'
3. $y' = x\sin\alpha + y\cos\alpha = 4(1) + 0(0) = 4$. plus on y'

The image is $(0, 4)$ – the point swings a quarter turn up, exactly as in the figure.

Example 5

Rotate the point $(3, 1)$ anticlockwise by 90°.

SOLUTION

1. $\cos 90^\circ = 0$, $\sin 90^\circ = 1$.
2. $x' = 3(0) - 1(1) = -1$.
3. $y' = 3(1) + 1(0) = 3$. 90°: x,y → -y,x

The image is $(-1, 3)$. (A neat pattern: a 90° turn sends (x, y) to $(-y, x)$.)

Example 6

Rotate the point $(4, 0)$ by 180° (a half-turn).

SOLUTION

1. $\cos 180^\circ = -1$, $\sin 180^\circ = 0$. half-turn values
2. $x' = 4(-1) - 0(0) = -4$.
3. $y' = 4(0) + 0(-1) = 0$. 180°: x,y → -x,-y

The image is $(-4, 0)$ – a half-turn flips a point straight through the origin to the opposite side.

Example 7

Rotate the point $(1, 1)$ anticlockwise by 45°.

SOLUTION

1. The 45° values are equal: $\cos 45^\circ = \sin 45^\circ \approx 0.707$. *45°: cos = sin*
2. $x' = x\cos\alpha - y\sin\alpha = 1(0.707) - 1(0.707) = 0$. *the parts cancel*
3. $y' = x\sin\alpha + y\cos\alpha = 1(0.707) + 1(0.707) \approx 1.41$.

The image is about $(0,\ 1.41)$ – the point sat on the diagonal, so a 45° turn stands it straight up onto the y-axis, still distance $\sqrt{2} \approx 1.41$ from the origin.



Rotating (1, 1) by 45° anticlockwise. The point lay on the diagonal, so the quarter-of-a-quarter turn lifts it onto the y-axis, keeping its length √2 ≈ 1.41.

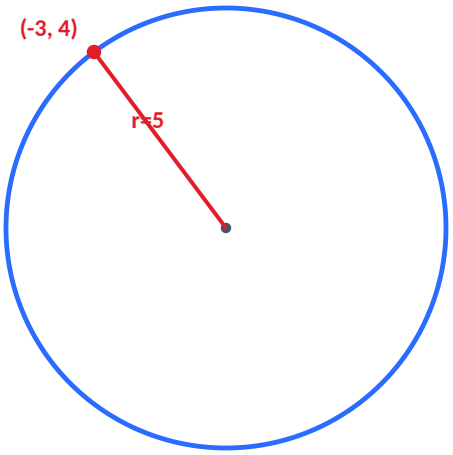
Example 8

Convert the Cartesian point $(-3, 4)$ (second quadrant) to polar.

SOLUTION

1. Distance first: $r = \sqrt{(-3)^2 + 4^2} = \sqrt{9 + 16} = \sqrt{25} = 5$. *squares kill the sign*
2. Angle: $\theta = \operatorname{atan2}(4, -3)$. Here x is negative but y is positive, so the point is in the **second quadrant**. *check the quadrant*
3. A plain $\arctan\frac{4}{-3} \approx -53^\circ$ would wrongly point into the fourth quadrant; $\operatorname{atan2}$ corrects it by adding 180° : $-53^\circ + 180^\circ = 127^\circ$.

The polar form is $(5,\ 127^\circ)$ – same distance 5 as the first-quadrant cousin $(3, 4)$, but aimed up and to the left.



The point (-3, 4) sits in the second quadrant. Its distance is still r = 5, but its direction angle is θ ≈ 127°, measured anticlockwise from the right – exactly what atan2 returns.

Example 9

Rotate the point $(5, 2)$ anticlockwise by 90° about the **centre** $(1, 1)$, not the origin.

SOLUTION

1. The rotation formulas only turn about the origin, so first **shift** the centre to the origin by subtracting it: $(5, 2) - (1, 1) = (4, 1)$. *move centre to 0*
2. Now rotate $(4, 1)$ by 90°. Using $(x, y) \to (-y, x)$: $(4, 1) \to (-1, 4)$. *quarter-turn shuffle*
3. Finally **shift back** by adding the centre again: $(-1, 4) + (1, 1) = (0, 5)$. *add the centre back*

The image is $(0, 5)$. To rotate about any centre: subtract it, rotate about the origin, then add it back – three tidy steps.

Note: All these rotation formulas assume the angle is measured **anticlockwise** (the positive direction). To rotate **clockwise**, use a negative angle, e.g. -90°; the formulas then handle the signs for you.

Mental-Math Shortcut

You can rotate by the three "quarter" angles without any cosine at all, just by shuffling coordinates and flipping signs: - 90° **anticlockwise:** $(x, y) \to (-y, x)$. - 180°**:** $(x, y) \to (-x, -y)$ – just negate both. - 270° **anticlockwise** (same as 90° clockwise): $(x, y) \to (y, -x)$.

Sanity-check with $(3, 1)$: a 90° turn gives $(-1, 3)$ ✓, another 90° (now 180°) gives $(-3, -1)$ ✓. And for **polar**, a rotation is the easiest move of all: just **add the angle to θ and leave r alone**. Spinning a sprite is a single addition – that is exactly why games store angles, not slopes.

🧠 The Logic of It

Why does a rotation leave the distance r unchanged while the formulas mix x and y together? Reason it from the polar picture. A point at distance r and angle θ is, by the conversion rules,

$$x = r\cos\theta, \qquad y = r\sin\theta.$$

Rotating by α means one thing only: keep r, replace θ by $\theta + \alpha$. So the new point is

$$x' = r\cos(\theta + \alpha), \qquad y' = r\sin(\theta + \alpha).$$

Now apply the angle-sum identities, $\cos(\theta + \alpha) = \cos\theta\cos\alpha - \sin\theta\sin\alpha$ and $\sin(\theta + \alpha) = \sin\theta\cos\alpha + \cos\theta\sin\alpha$, and substitute back $r\cos\theta = x$, $r\sin\theta = y$:

$$x' = x\cos\alpha - y\sin\alpha, \qquad y' = x\sin\alpha + y\cos\alpha.$$

The Cartesian rotation formulas were never an arbitrary recipe – they are the polar statement "add α to the angle, keep the radius" translated, by *if–then* substitution, into x-and-y language. Because r is untouched at every step, every rotation is a rigid turn: distances to the origin are preserved, which is precisely what we mean by "rotate".

⚠ Common Mistakes

- Mixing up the rotation formulas: $x' = x\cos\alpha - y\sin\alpha$ uses a **minus**, while $y' = x\sin\alpha + y\cos\alpha$ uses a **plus**. Swapping the signs rotates the wrong way.
- Using plain $\arctan(y/x)$ for the polar angle and getting the wrong quadrant. Use $\operatorname{atan2}(y, x)$, which checks both signs.
- Putting r and θ in the wrong order, or assuming the angle is in degrees when a tool expects radians. State your units and convert deliberately.
- Forgetting that these formulas rotate about the **origin** only. To turn about another centre, first subtract that centre, rotate, then add it back.
- Rotating clockwise with a positive angle. Clockwise is the **negative** direction: use $-\alpha$.

Quiz

43.1.Q1. In polar coordinates (r, θ), what does r represent?

A. The angle

B. The distance from the origin

C. The x-coordinate

D. The slope

43.1.Q2. Which formula gives the x-coordinate from polar form?

A. $x = r \sin \theta$

B. $x = r \cos \theta$

C. $x = r + \theta$

D. $x = r / \theta$

43.1.Q3. Convert the polar point $(2, 0^\circ)$ to Cartesian.

43.1.Q4. Find the polar distance r for the Cartesian point $(6, 8)$.

A. 10

B. 14

C. 48

D. 7

43.1.Q5. Rotate $(1, 0)$ anticlockwise by 90°. Where does it land?

A. $(0, 1)$

B. $(-1, 0)$

C. $(0, -1)$

D. $(1, 0)$

43.1.Q6. True or False: rotating a point about the origin changes its distance from the origin.

A. True

B. False

43.1.Q7. Rotate $(2, 0)$ by 180°.

43.1.Q8. Why use $\operatorname{atan2}(y, x)$ instead of $\arctan(y/x)$ for the polar angle?

🐍 From Concept to Code (Python)

These conversions and rotations are short, exact functions in Python's `math` module. Angles in code are **radians**, so we convert at the door with `radians` and `degrees`. Below, a point round-trips polar → Cartesian → polar (and lands back where it began), then we write a reusable `rotate`.

```
import math

def to_cartesian(r, deg):
    a = math.radians(deg)
    return (round(r * math.cos(a), 2), round(r * math.sin(a), 2))

def to_polar(x, y):
    return (round(math.hypot(x, y), 2),
            round(math.degrees(math.atan2(y, x)), 2))

print(to_cartesian(5, 53.13))   # distance 5, angle ~53 deg
print(to_polar(3, 4))           # ...and back again
```

```
output
(3.0, 4.0)
(5.0, 53.13)
```

```
import math

def rotate(x, y, deg):
    a = math.radians(deg)
    xr = x * math.cos(a) - y * math.sin(a)
    yr = x * math.sin(a) + y * math.cos(a)
    return (round(xr, 2), round(yr, 2))

print(rotate(4, 0, 90))    # quarter turn up
print(rotate(3, 1, 90))    # (x, y) -> (-y, x)
print(rotate(4, 0, 180))   # half turn
```

```
output
(0.0, 4.0)
(-1.0, 3.0)
(-4.0, 0.0)
```

Try it:

1. Call `rotate(1, 0, 45)` and confirm you get about `(0.71, 0.71)` – a point on the diagonal.
2. Rotate `(2, 0)` by `90` four times in a row (feeding each result into the next). You should return to `(2, 0)`, having gone all the way around.

Answer Key – Chapter 43 – Rotations & Polar Coordinates

43.1.Q1. The distance from the origin (the pole); θ is the direction angle.

43.1.Q2. $x = r\cos\theta$ – cosine handles the horizontal part.

43.1.Q3. $(2, 0)$ – at angle 0°, $x = 2\cos 0^\circ = 2$ and $y = 2\sin 0^\circ = 0$.

43.1.Q4. 10 – $r = \sqrt{6^2 + 8^2} = \sqrt{100} = 10$.

43.1.Q5. $(0, 1)$ – a quarter-turn sends (x, y) to $(-y, x)$, so $(1, 0) \to (0, 1)$.

43.1.Q6. False – a rotation keeps the distance r the same; only the direction changes.

43.1.Q7. $(-2, 0)$ – a half-turn negates both coordinates: $(x, y) \to (-x, -y)$.

43.1.Q8. Because $\mathrm{atan2}$ checks the signs of x and y and returns the angle in the correct quadrant; plain $\arctan$ cannot tell the quadrants apart.



PART IX

Geometry

Shape, size, position and space. The building blocks – points, lines, angles, triangles and circles – through coordinate geometry, transformations, 3-D shapes, and the intersection tests that power graphics and games.

Chapter 44 – Points, Lines & Planes · Chapter 45 – Angles · Chapter 46 – Triangles · Chapter 47 – Circles · Chapter 48 – Coordinate Geometry · Chapter 49 – Distance, Area & Volume · Chapter 50 – Transformations & Geometric Vectors · Chapter 51 – Projections · Chapter 52 – Intersection & Collision Geometry · Chapter 53 – Graphs & Networks

Chapter 44 · Points, Lines & Planes

The four building blocks of all geometry – the point, the line, the ray and the segment – how to name them, when points are collinear, and the difference between lines that meet and lines that never do.

The Idea Behind It

Through two distinct points there is exactly one straight line, never two. but why?

Lay a straightedge so it touches two dots: you can find *a* line easily enough. The real claim is that you can never find a *second*, different one. The proof leans on the one property that makes a line a line:

$$\boxed{\text{a straight line never bends: fix two of its points and the whole line is fixed}}$$

Suppose, for contradiction, that **two different** lines both passed through the same points A and B. Compare them point by point along the way from A to B and beyond:

$$\text{line 1 and line 2 agree at } A \text{ and at } B$$

Two straight paths that already meet at A and meet again at B cannot peel apart anywhere, because peeling apart would mean bending, and a line cannot bend. So they coincide at every point: they are the same line, contradicting *"different"*.

$$\therefore \text{ the line through } A \text{ and } B \text{ is unique}$$

"Exactly one" packs two facts together: at least one line exists (the straightedge finds it) **and** at most one does (the argument above). this is why we may say "the" line through two points

Every drawing in geometry, every shape on a screen, every map and floor-plan, is built from a tiny handful of pieces. Before we can talk about triangles or circles we need those pieces. They are so simple we never *define* them with other words – we just point at them and agree what they are. This chapter is the alphabet of geometry: learn these letters and every later shape becomes a word you can read.

The point: a position with no size

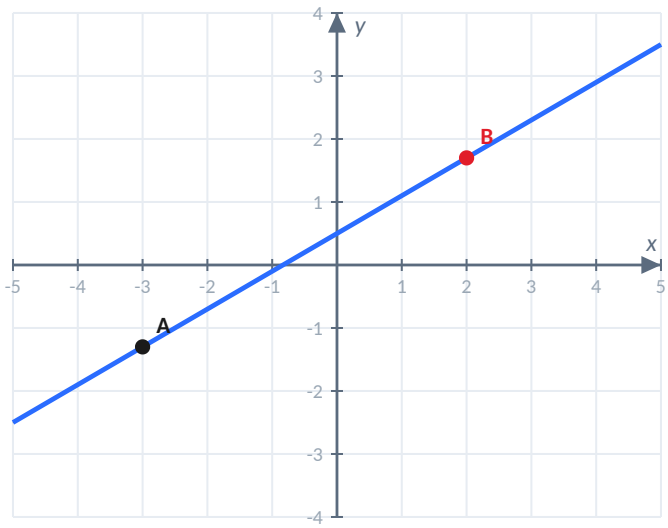
A **point** marks a single position – nothing more. It has no width, no height, no thickness. On paper we draw a dot, but the real point is the *exact spot at its centre*. We name a point with a single capital letter, like A.

POINT

A **point** is an exact location in space. It has **no size** – no length, width or depth – only a position. We label it with a capital letter, e.g. point A.

The line: straight, and endless both ways

Slide a point and the trail it leaves is a **line**. A line is perfectly straight and runs forever in *both* directions, which we show by drawing a small arrowhead on each end. Two points are enough to fix a line in place, so we name a line by any two of its points with a two-headed arrow above them, like $\overleftrightarrow{AB}$.



The line AB (blue) passes through points A and B and continues forever past both arrowheads.

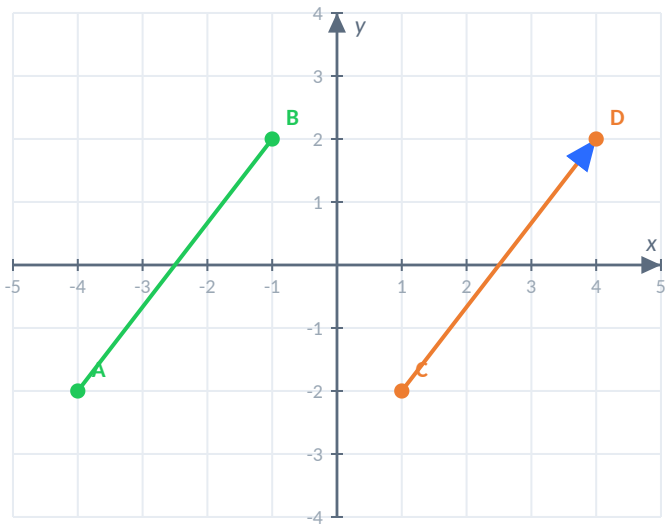
LINE

A **line** is a straight path that extends **without end** in both directions. It has length but no thickness. Through any two points there is exactly one line; we write it $\overleftrightarrow{AB}$.

Segments and rays: pieces of a line

Most of the time we only want *part* of a line. Cut a line at two points and keep the piece between them and you have a **line segment** – it has two endpoints and a definite length. Cut a line at one point and keep everything on one side and you have a **ray** – one endpoint, then off to infinity (a beam of light from a torch is a good picture of a ray).

$$\underset{\text{segment}}{\overline{AB}} \text{ has two ends.} \qquad \underset{\text{ray}}{\overrightarrow{AB}} \text{ starts at } A\text{, passes } B\text{, never stops.}$$



Left: segment AB (green) stops at both ends. Right: ray CD (orange) starts at C and shoots off past D.

LINE SEGMENT

A **line segment** $\overline{AB}$ is the part of a line **between** two endpoints A and B, including those endpoints. It has a fixed length.

RAY

A **ray** $\overrightarrow{AB}$ starts at an endpoint A, passes through B, and continues **forever** in that one direction.

point line line segment ray collinear plane

Collinear points and the flat plane

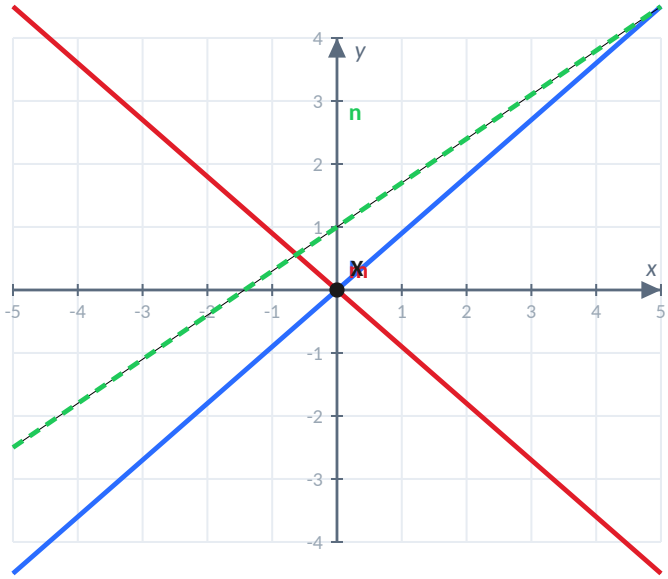
Points that all sit on **one** straight line are called **collinear**. Any *two* points are automatically collinear (you can always draw a line through two dots). The interesting question is whether a *third* point joins them on that same line, or sits off it.

Finally, a **plane** is a perfectly flat surface – like an infinite tabletop or sheet of glass – stretching forever in every direction. Points and lines that all lie on one plane are **coplanar**. A flat screen, a wall, a page: each is a everyday piece of a plane.

When lines meet – and when they do not

Two different lines in the same plane do one of just two things:

- **Intersecting lines** cross at exactly **one** point.
- **Parallel lines** run in the same direction and **never** meet, no matter how far you extend them (think of the two rails of a train track).



Lines l and m intersect at the single point X. Line n is parallel to l (same slope), so n and l never meet.

Worked examples

Example 1

How many different lines can be drawn through two distinct points P and Q?

SOLUTION

1. Place the two points. A line must be straight. *2 points fix a line*
2. Lay a straightedge so it touches both P and Q – there is only one way to do this.
3. Any other line through P would tilt away and miss Q.

Exactly one line. "Two points determine a line" is a foundational rule of geometry.

Example 2

A line passes through $A(0,0)$ and $B(4,2)$. Is the point $C(6,3)$ collinear with A and B?

SOLUTION

1. Find how the line climbs from A to B: rise over run is $\frac{2}{4} = \frac{1}{2}$. *rise/run*
2. Test C: from $A(0,0)$ to $C(6,3)$ the climb is $\frac{3}{6} = \frac{1}{2}$ – the same. *same slope*
3. Equal steepness from the same starting point means C lies on the very same straight line.

Yes – A, B and C are collinear (all three sit on the line of slope $\frac{1}{2}$ through the origin).

Example 3

Name every line segment that can be formed from three collinear points A, B, C (in that order).

SOLUTION

1. A segment is named by its two endpoints, and $\overline{AB}$ is the same segment as $\overline{BA}$. *order ignored*
2. Pick pairs of points: A with B, A with C, B with C.

Three segments: $\overline{AB}$, $\overline{BC}$ and $\overline{AC}$.

Example 4

A ray $\overrightarrow{AB}$ and a ray $\overrightarrow{BA}$ share the same two points. Are they the same ray?

SOLUTION

1. $\overrightarrow{AB}$ starts at A and heads out past B.
2. $\overrightarrow{BA}$ starts at B and heads out past A – the opposite direction. *other way*
3. They overlap on the segment between A and B, but point opposite ways.

No. A ray is fixed by its starting point and direction, so $\overrightarrow{AB} \neq \overrightarrow{BA}$.

Example 5

Lines l and m lie in one plane. Line l has slope 3; line m has slope 3 but a different intercept. Do they intersect?

SOLUTION

1. Two lines meet where they would cross. Lines with the **same slope** rise at the same rate. *slopes match*
2. Because the intercepts differ, one line is always a fixed distance above the other.
3. They march along side by side and never close the gap.

No – equal slopes (and different intercepts) means the lines are parallel; they never intersect.

Example 6

How many points do two distinct intersecting lines share? How many do two distinct parallel lines share?

SOLUTION

1. Two different straight lines can cross at most once – beyond that they would have to be the same line.
2. Intersecting lines, by definition, do cross: that is 1 shared point. *cross once*
3. Parallel lines, by definition, never cross: 0 shared points.

Intersecting lines share exactly one point; parallel lines share none.

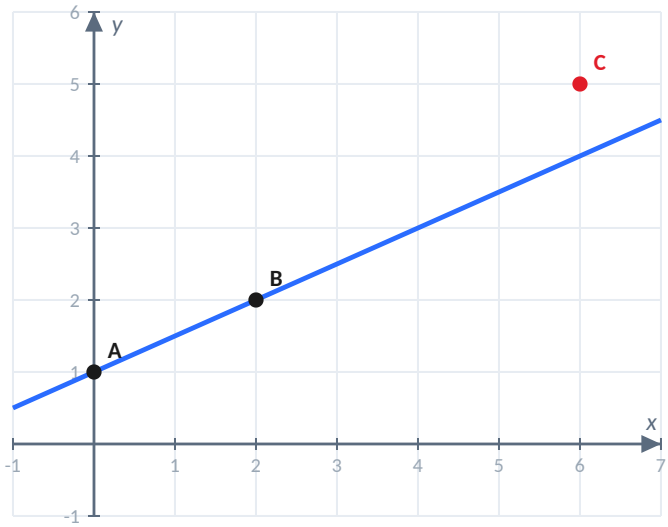
Example 7

A line passes through $A(0,1)$ and $B(2,2)$. Is the point $C(6,5)$ collinear with A and B?

SOLUTION

1. Climb from A to B: rise over run is $\frac{2-1}{2-0} = \frac{1}{2}$. rise/run
2. Climb from $A(0,1)$ to $C(6,5)$: rise over run is $\frac{5-1}{6-0} = \frac{4}{6} = \frac{2}{3}$. different slope
3. The two steepnesses, $\frac{1}{2}$ and $\frac{2}{3}$, are **not** equal, so C drifts off the line of A and B.

No – C is not collinear with A and B; it sits just above their line.



A and B fix the blue line (slope ½). Point C(6,5) lies above it – on the line it would be at (6,4) – so the three points are not collinear.

Example 8

Three points A, B and C are **not** all on one line. How many flat planes pass through all three?

SOLUTION

1. Two of the points, say A and B, fix a single line. 2 points → 1 line
2. The third point C lies **off** that line, so it pins down which way the flat surface tilts.
3. Tilt a plane to contain the line AB and you can still spin it like a page on a spine – until you also force it through C, which locks it in exactly 1 position.

Exactly one plane. "Three non-collinear points determine a plane" – the flat-surface twin of "two points determine a line" (it is why a three-legged stool never wobbles).

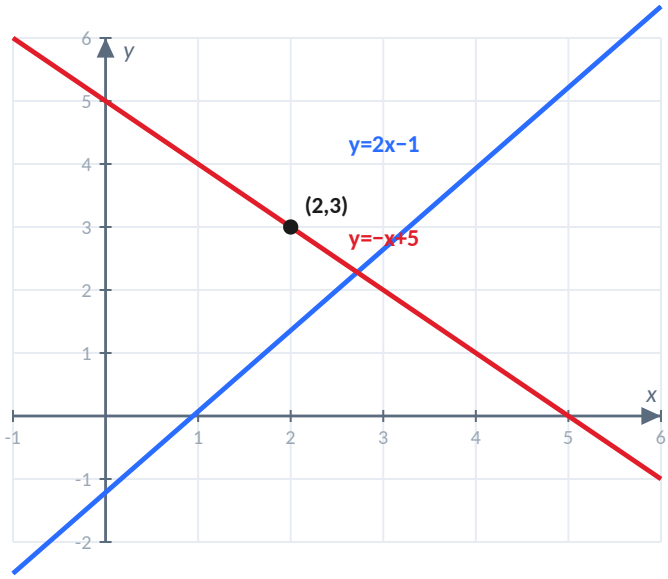
Example 9

Two lines are $y = 2x - 1$ and $y = -x + 5$. They have different slopes, so they meet. Find the point where they intersect.

SOLUTION

1. At the crossing point both equations give the **same** y, so set them equal: $2x - 1 = -x + 5$.
   same point → same y
2. Gather the x-terms: $2x + x = 5 + 1$, so $3x = 6$, giving $x = 2$. solve for x
3. Put $x = 2$ back into either line: $y = 2(2) - 1 = 3$.

The lines cross at the single point $(2, 3)$ – the one place where both equations are true at once.



Two lines with different slopes meet at exactly one point. Solving 2x−1 = −x+5 pins it down at (2, 3).

Note: Capital letters name points; a small letter (like l) or two points name a line. Watch the symbol above two letters: a bar $\overline{AB}$ is a **segment**, a single arrow $\overrightarrow{AB}$ is a **ray**, and a double arrow $\overleftrightarrow{AB}$ is the full **line**.

Mental-Math Shortcut

To test whether a third point is **collinear** with a line through two known points, compare "rise over run" without any formula. Find the climb from the first point to the second (down and across), reduce it to a simple fraction, then do the same from the first point to the new point. **Same fraction → same line.** For $A(0, 0)$ and $B(3, 6)$ the climb is $\frac{6}{3} = 2$; a point like $(5, 10)$ gives $\frac{10}{5} = 2$, so it is on the line, while $(5, 9)$ gives $\frac{9}{5} \neq 2$, so it is off it. You can often see "same steepness" at a glance, no arithmetic needed.

One of the deepest facts in geometry is also one of the simplest: **two distinct points determine exactly one line.** This is an **axiom** – a starting assumption we accept without proof because everything else is built on it – but we can still reason about *why* it is the right assumption, and it splits neatly into two claims joined by an AND.

First, **at least one** line exists through points A and B: lay a straightedge touching both, and there is your line. Second, there is **at most one**: suppose two different lines both passed through A and B. They would share those two points yet head off in different directions – but a straight line cannot bend, so once two lines agree at two separate points they must agree everywhere and are really the *same* line, a contradiction.

"Exactly one" is shorthand for **"at least one" AND "at most one"** – existence and uniqueness together. That two-part pattern (something exists, *and* nothing else like it does) is one of the most common shapes a mathematical statement can take, and you will meet it again every time the word "the" appears: *the* midpoint, *the* solution, *the* line through two points.

⚠ Common Mistakes

- Treating a line segment and a line as the same thing. A **segment** stops at two endpoints; a **line** has arrowheads and runs forever.
- Thinking $\overrightarrow{AB}$ and $\overrightarrow{BA}$ are the same ray. They start at different points and point in opposite directions.
- Believing two points could have several lines through them. There is always **exactly one** line through two distinct points.
- Calling three points collinear just because you *drew* them roughly in a row. Collinear means they lie on one **exact** straight line – check the steepness.
- Confusing parallel with merely "not yet crossed". Parallel lines have the same direction and **never** meet, even if extended forever.

Quiz

44.1.Q1. How many distinct lines pass through two different points?

A. Zero

B. Exactly one

C. Exactly two

D. Infinitely many

44.1.Q2. Which figure has exactly two endpoints?

A. A line

B. A ray

C. A line segment

D. A plane

44.1.Q3. A ray has how many endpoints?

A. 0

B. 1

C. 2

D. Infinitely many

44.1.Q4. Points that all lie on one straight line are called ___.

44.1.Q5. Two distinct lines in a plane that never meet are called ___.

A. Intersecting

B. Perpendicular

C. Parallel

D. Collinear

44.1.Q6. At how many points do two distinct intersecting lines cross?

A. 0

B. 1

C. 2

D. Infinitely many

44.1.Q7. True or False: a point has length but no width.

A. True

B. False

44.1.Q8. The symbol $\overline{PQ}$ refers to what?

A. A line

B. A ray

C. A segment

D. A plane

44.1.Q9. Are the points $A(0,0)$, $B(2,4)$ and $C(3,6)$ collinear?

🐍 From Concept to Code (Python)

In code a **point** is just its two coordinates bundled together – a *tuple* `(x, y)`. From two points we can test the most basic question of this chapter: are three points collinear? They are exactly when the climb (slope) from the first to the second matches the climb from the first to the third. To dodge dividing by zero we compare the slopes by **cross-multiplying** instead.

```
def collinear(a, b, c):
    # a, b, c are points (x, y). Same slope from a means one line.
    # (by-bx) ... compare rise*run cross-products
    return (b[1]-a[1]) * (c[0]-a[0]) == (c[1]-a[1]) * (b[0]-a[0])

A = (0, 0)
B = (2, 4)
print(collinear(A, B, (3, 6)))   # on the line of slope 2?
print(collinear(A, B, (3, 5)))   # off it?
```

```
output
True
False
```

Try it:

1. Store a point as `p = (4, 7)` and print its x with `p[0]` and its y with `p[1]`.
2. Use `collinear` to check whether `(0,0)`, `(1,1)` and `(5,5)` lie on one line. Then try `(5,6)`.

Answer Key – Chapter 44 – Points, Lines & Planes

44.1.Q1. Exactly one – two points determine a single line.

44.1.Q2. A line segment – it is the part of a line between (and including) two endpoints.

44.1.Q3. 1 – a ray starts at one endpoint and continues forever in one direction.

44.1.Q4. Collinear – "co" (together) on one "line".

44.1.Q5. Parallel – same direction, no shared point.

44.1.Q6. 1 – two different straight lines can share at most one point, and intersecting lines share exactly one.

44.1.Q7. False – a point has **no** size at all, neither length nor width; it is only a position.

44.1.Q8. A segment – the bar (no arrowheads) means the piece between endpoints P and Q.

44.1.Q9. Yes – each gives the same steepness $\frac{y}{x} = 2$ from the origin, so all three lie on one line.

Chapter 45 · Angles

What an angle measures, how to name the five kinds by size, the partner-pairs that add to 90° and 180°, and the equal angles that appear whenever a line crosses two parallel lines.

The Idea Behind It

When two lines cross, the angles directly opposite each other are always equal. *but why?*

Cross two straight lines and four angles appear. The two facing across the crossing look equal, and they truly are, every time. No protractor is needed: one fact and a cancellation settle it.

$$\boxed{\text{angles in a row on a straight line add to } 180^\circ}$$

Label two angles that sit *next to each other* as a and b, and the angle on the far side of b as c. Each neighbouring pair lies along one straight line, so each pair is supplementary:

$$a + b = 180^\circ \qquad \text{and} \qquad b + c = 180^\circ$$

Both sums equal 180°, so they equal each other; now subtract the shared b from both sides and it cancels:

$$a + b = c + b \implies a = c$$

But a and c are exactly the pair sitting **opposite** across the crossing.

$$\therefore \text{ vertically opposite angles are } \text{equal}$$

Each is "180° minus the same neighbour", so they must match.

two things equal to the same thing are equal to each other

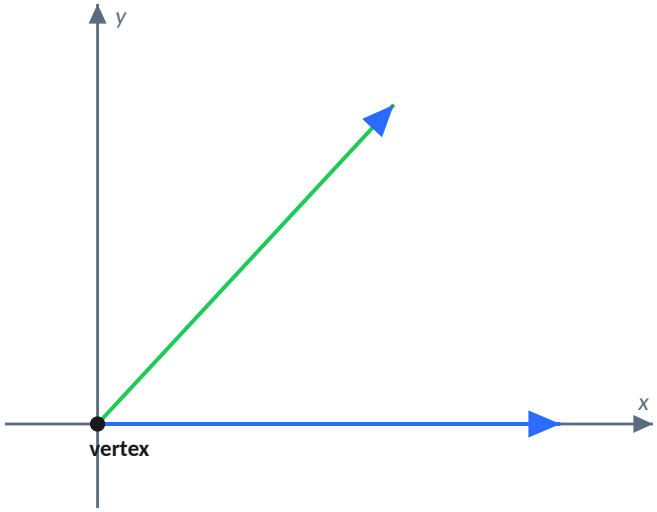
Stand still and turn to face a new direction: the amount you turned is an **angle**. Angles are everywhere – the corner of a page, the spread of a pair of scissors, the tilt of a ramp, the direction you steer. This chapter measures that turning, sorts angles into a few simple types, and uncovers the neat relationships that let you find an unknown angle without ever picking up a protractor.

What an angle is

An **angle** is formed by two **rays** sharing a single starting point. That shared point is the **vertex**; the two rays are the **arms**. We measure the *amount of turn* between the arms in **degrees** (°), where a full turn all the way round is 360°.

ANGLE

An **angle** is the amount of turn between two **rays** that meet at a common point, the **vertex**. It is measured in **degrees**; one full revolution is 360°.

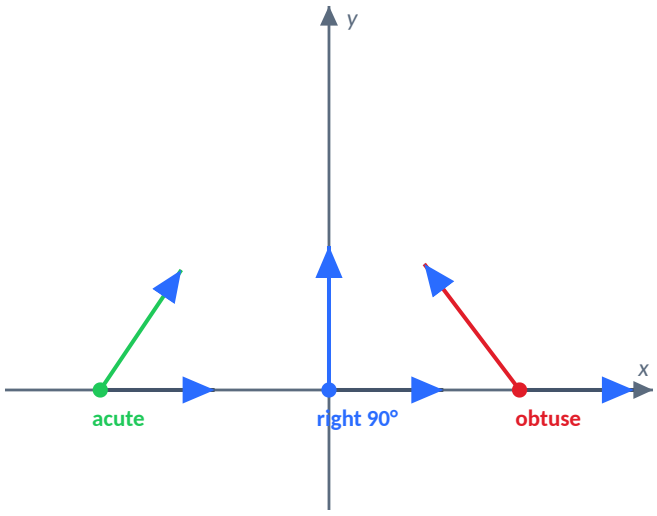


An angle: two rays (arms) from one vertex. The angle is the turn from the blue arm round to the green arm.

The five types, by size

Sort angles by how big the turn is and you get five named types. Two special markers anchor them: a **right angle** is a quarter-turn (90°, drawn with a small square), and a **straight angle** is a half-turn (180°, a straight line).

- **Acute**: less than 90° (a sharp corner).
- **Right**: exactly 90° (a perfect square corner).
- **Obtuse**: between 90° and 180° (a blunt, open corner).
- **Straight**: exactly 180° (the two arms form a straight line).
- **Reflex**: more than 180° but less than 360° (a turn the "long way round").



Three of the five types: an acute angle (under 90°), a right angle (exactly 90°), and an obtuse angle (between 90° and 180°).

angle vertex complementary supplementary vertically opposite transversal

Partner pairs: complementary and supplementary

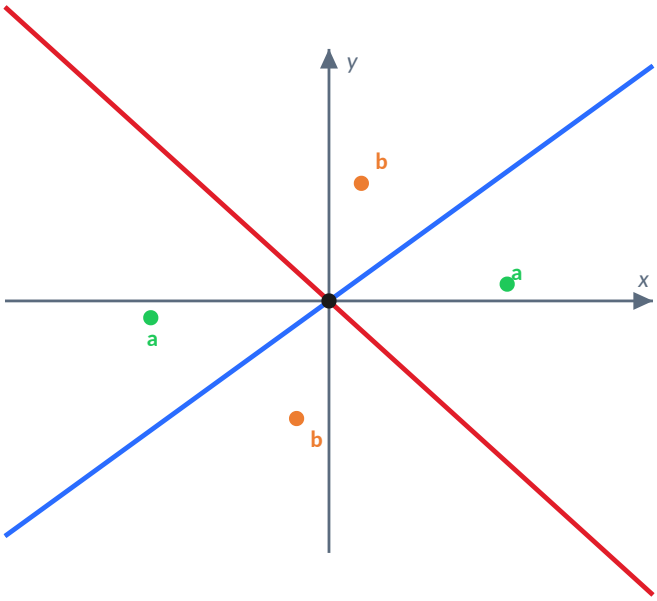
Two angles are **complementary** when they add up to 90°, and **supplementary** when they add up to 180°. If you know one angle of such a pair, a single subtraction gives the other.

$$\text{complementary: } a + b = 90^\circ \qquad \text{supplementary: } a + b = 180^\circ$$

Tip: Remember the order alphabetically: **c**omplementary comes before **s**upplementary, and 90 comes before 180. So **c → 90**, **s → 180**.

Vertically opposite angles

When two straight lines cross, they make four angles. The two angles **directly across** the vertex from each other are always **equal** – these are **vertically opposite** angles. (The two angles next to each other on a straight line are supplementary, which is why the opposite ones come out equal.)

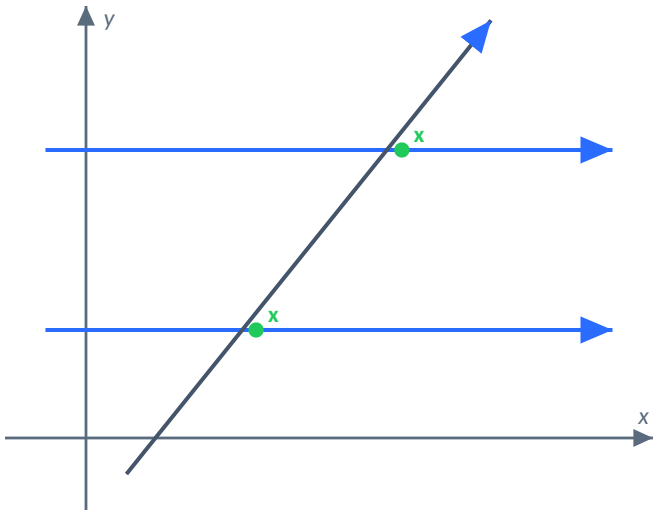


Two crossing lines make four angles. The pairs directly opposite the crossing (both a, both b) are equal: vertically opposite angles.

Angles with parallel lines

Now draw a third line, a **transversal**, cutting across two **parallel** lines. Eight angles appear, and they come in matching families. Three pairings matter:

- **Corresponding** angles (same corner position at each crossing) are **equal** – they form an "F" shape.
- **Alternate** angles (opposite sides of the transversal, between the parallels) are **equal** – they form a "Z" shape.
- **Co-interior** angles (same side, between the parallels) are **supplementary**, adding to 180° – they form a "C" shape.



A transversal (grey) crosses two parallel lines (blue). The two marked angles x sit in the same corner at each crossing – corresponding angles, so they are equal.

Worked examples

Example 1

Classify a 135° angle.

SOLUTION

1. Compare with the two markers: 90° (right) and 180° (straight).
2. 135° is bigger than 90° but smaller than 180°. between 90 and 180

It is an obtuse angle.

Example 2

An angle is complementary to 32°. Find it.

SOLUTION

1. Complementary angles add to 90°. c → 90
2. Subtract the known angle: $90^\circ - 32^\circ = 58^\circ$.

The complement is 58°.

Example 3

An angle is supplementary to 110°. Find it.

SOLUTION

1. Supplementary angles add to 180°. *s → 180*
2. Subtract: $180^\circ - 110^\circ = 70^\circ$.

The supplement is 70°.

Example 4

Two lines cross. One of the four angles is 63°. Find the other three.

SOLUTION

1. The angle **vertically opposite** the 63° is equal to it: 63°. *opposite = equal*
2. Each angle **next to** the 63° on a straight line is supplementary: $180^\circ - 63^\circ = 117^\circ$.
3. The fourth angle is vertically opposite a 117°, so it is 117° too.

The four angles are $63^\circ, 117^\circ, 63^\circ, 117^\circ$.

Example 5

A transversal crosses two parallel lines. A corresponding angle measures 74°. What is its partner at the other crossing?

SOLUTION

1. Corresponding angles (the matching "F"-shape corners) are equal when the lines are parallel. *F = equal*
2. So the partner equals the given angle.

The corresponding angle is 74°.

Example 6

A transversal crosses two parallel lines. A co-interior angle is 118°. Find the co-interior angle paired with it.

SOLUTION

1. Co-interior angles (the "C"-shape, same side and between the parallels) are supplementary. *C → 180*
2. Subtract from 180°: $180^\circ - 118^\circ = 62^\circ$.

The paired co-interior angle is 62°.

Example 7

Angles A and B are complementary, and A is twice B. Find both.

SOLUTION

1. Write the condition: $A + B = 90^\circ$ and $A = 2B$.
2. Substitute $A = 2B$ into the sum: $2B + B = 90^\circ$, so $3B = 90^\circ$. *sub in*
3. Then $B = 30^\circ$, and $A = 2(30^\circ) = 60^\circ$.

$A = 60^\circ$ **and** $B = 30^\circ$ **(and indeed** $60 + 30 = 90$**).**

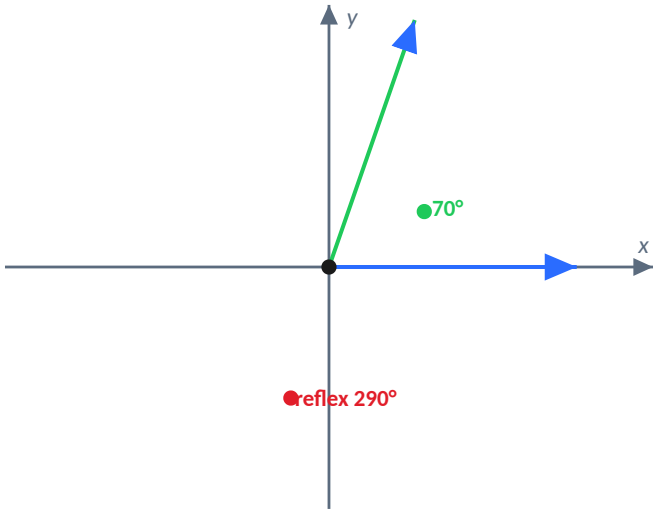
Example 8

Find the **reflex** angle that goes with a 70° angle.

SOLUTION

1. The angle and its reflex together make one **full turn**, which is 360°. *reflex + angle = 360*
2. So the reflex is a full turn minus the angle: $360^\circ - 70^\circ = 290^\circ$.

The reflex angle is 290° **– the "long way round" past** 180°**, not the** 110° **many people guess.**



The narrow 70° turn between the arms, and its reflex 290° measured the long way round the outside. Together they complete a full 360° turn.

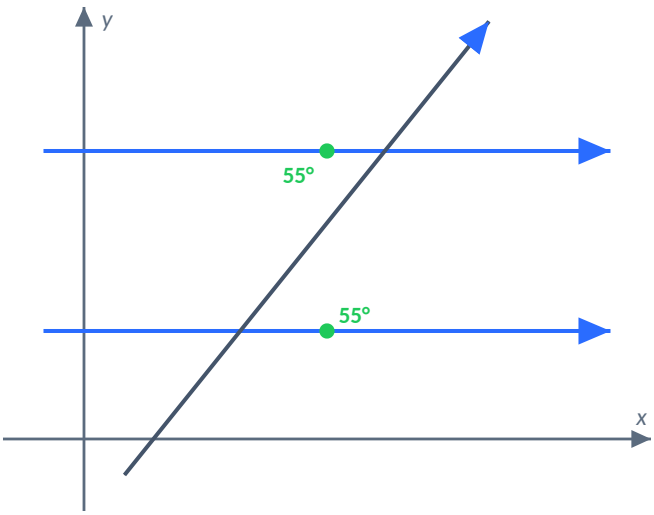
Example 9

A transversal crosses two parallel lines. An alternate angle ("Z"-shape) is 55°. Find its partner across the transversal.

SOLUTION

1. Alternate angles lie on **opposite sides** of the transversal, tucked between the two parallels – the "Z" shape.
   Z = equal
2. When the lines are parallel, alternate angles are **equal**, so the partner matches the given angle exactly.

The alternate angle is 55°.



A transversal (grey) crosses two parallel lines (blue). The two marked angles sit on opposite sides of the transversal, between the parallels – alternate ("Z") angles, so they are equal.

Example 10

Two angles are **supplementary**, and the larger is 30° more than the smaller. Find both angles.

SOLUTION

1. Call the smaller angle x; then the larger is $x + 30^\circ$, and supplementary means they add to 180°: $x + (x + 30^\circ) = 180^\circ$. *set up the sum*
2. Collect the terms: $2x + 30^\circ = 180^\circ$, so $2x = 150^\circ$ and $x = 75^\circ$. *solve for x*
3. The larger is $x + 30^\circ = 75^\circ + 30^\circ = 105^\circ$.

The angles are 75° and 105° (and indeed $75 + 105 = 180$).

Note: Vertically opposite angles are always equal, but the special parallel-line rules (corresponding equal, alternate equal, co-interior supplementary) only hold when the two lines really are **parallel**. If the lines are not parallel, those equalities break.

⚡ Mental-Math Shortcut

To find a **complement**, subtract from 90; for a **supplement**, subtract from 180. The fast way uses *complements of the digits* – think "how far to the next round number". For the supplement of 137°: the units digit 7 needs 3 to reach 10, then the tens 13 need 4 to reach 17... simpler still, do $180 - 137$ as $180 - 140 + 3 = 43$. For a complement like $90 - 26$: take it to $90 - 30 + 4 = 64$. Round the awkward number to a neat one, subtract, then adjust – far quicker than borrowing.

Why are **vertically opposite angles always equal**? We can prove it with one fact and a cancellation – no measuring required. When two lines cross they make four angles; call two neighbours a and b, where a sits between the two arms and b is right beside it.

Because a and b together lie along one *straight* line, they are supplementary:

$$a + b = 180^\circ.$$

Now look at b and the angle on its *other* side, call it c. They also sit on a straight line, so

$$b + c = 180^\circ.$$

Both sums equal 180°, so they equal each other: $a + b = b + c$. Subtract the shared b from both sides and it cancels, leaving

$$a = c.$$

But a and c are precisely the pair sitting **opposite** each other across the crossing. So the vertically opposite angles must be equal – not by luck, but because each is the "180° minus the same neighbour". That move, *two things equal to the same thing are equal to each other*, is one of logic's oldest tools, and here it turns a picture into a certainty.

⚠ Common Mistakes

- Swapping complementary and supplementary. **C**omplementary → 90°, **s**upplementary → 180° (alphabetical: c before s, 90 before 180).
- Assuming any two angles that "look" equal are vertically opposite. They must be the pair **directly across** the crossing of two straight lines.
- Using the parallel-line angle rules when the lines are **not** parallel. Corresponding/alternate equality needs genuine parallel lines.
- Calling an angle "right" just because it looks like a corner. A right angle is **exactly** 90°; check for the small square marker.
- Forgetting that a reflex angle is more than 180°: the reflex of a 70° angle is $360^\circ - 70^\circ = 290^\circ$, not 110°.

Quiz

45.1.Q1. An angle of 90° is called ___.

A. Acute

B. Right

C. Obtuse

D. Straight

45.1.Q2. Classify an angle of 200°.

A. Obtuse

B. Straight

C. Reflex

D. Acute

45.1.Q3. What is the complement of 25°?

45.1.Q4. What is the supplement of 145°?

45.1.Q5. Two lines cross; one angle is 40°. What is the angle vertically opposite it?

A. 40°

B. 50°

C. 140°

D. 320°

45.1.Q6. A transversal crosses two parallel lines. Alternate angles are ___.

A. Equal

B. Supplementary

C. Complementary

D. Always 90°

45.1.Q7. Co-interior angles between two parallel lines add to ___.

A. 90°

B. 180°

C. 360°

D. They are equal

45.1.Q8. True or False: an obtuse angle is larger than 180°.

A. True

B. False

45.1.Q9. Angles A and B are supplementary and equal. Find each.

🐍 From Concept to Code (Python)

Classifying an angle by its size is exactly the kind of decision `if / elif / else` was made for. We feed the program a number of degrees and it returns the type name – the same five buckets from the lesson, expressed as a ladder of comparisons.

```
def classify(deg):
    if deg == 90:
        return "right"
    elif deg == 180:
        return "straight"
    elif deg < 90:
        return "acute"
    elif deg < 180:
        return "obtuse"
    else:
        return "reflex"

for a in [30, 90, 135, 180, 250]:
    print(a, "->", classify(a))
```

```
output
30 -> acute
90 -> right
135 -> obtuse
180 -> straight
250 -> reflex
```

Try it:

1. Write `complement(a)` returning `90 - a` and `supplement(a)` returning `180 - a`. Test them on `25` and `145`.
2. Add a branch to `classify` so that an input of `360` (or `0`) returns `"full / zero turn"`.

Answer Key – Chapter 45 – Angles

45.1.Q1. Right – exactly a quarter-turn, 90°.

45.1.Q2. Reflex – it is more than 180° (but less than 360°).

45.1.Q3. 65° – complementary angles add to 90°, and $90 - 25 = 65$.

45.1.Q4. 35° – supplementary angles add to 180°, and $180 - 145 = 35$.

45.1.Q5. 40° – vertically opposite angles are equal.

45.1.Q6. Equal – alternate ("Z"-shape) angles between parallel lines are equal.

45.1.Q7. 180° – co-interior ("C"-shape) angles are supplementary.

45.1.Q8. False – obtuse is between 90° and 180°; larger than 180° would be reflex.

45.1.Q9. 90° each – they sum to 180° and are equal, so $180 \div 2 = 90$.

Chapter 46 · Triangles

How to name a triangle by its sides and angles, why the three angles always add to 180°, the Pythagoras rule for right triangles, two ways to find the area, and what makes two triangles congruent or similar.

The Idea Behind It

Every triangle's three angles add to exactly 180°, fat or thin, big or small. *but why?*

Tear the three corners off any paper triangle and lay them side by side: they always line up into one straight edge. That is 180°, and it is no coincidence. The proof borrows one rule from the previous chapter:

$$\boxed{\text{alternate angles on parallel lines are equal (the } Z \text{ shape)}}$$

Take a triangle with angles A, B, C, and through the top vertex draw a line **parallel** to the base. Each slanted side now cuts two parallel lines, so by the alternate-angles rule a copy of each base angle reappears at the top:

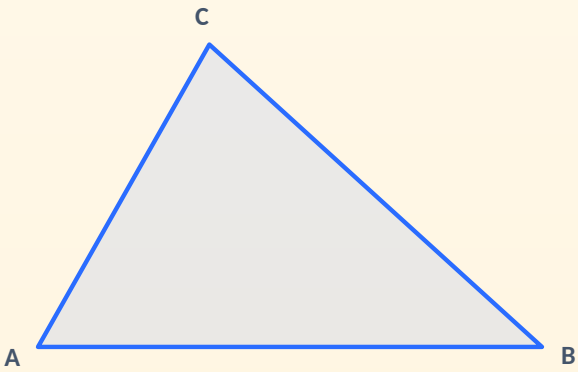
$$\text{left side gives } A\,, \qquad \text{right side gives } B$$

Now look along that straight parallel line through the top vertex. Three angles sit in a row there: the copy of A, then C at the vertex itself, then the copy of B. Angles in a row on a straight line make 180°:

$$A + C + B = 180^\circ$$

$$\therefore\ A + B + C = 180^\circ$$

The argument never used the triangle's size or shape, only "alternate angles are equal" and "a straight line is 180°", so it holds for **all** triangles at once. *proving a universal, not checking one example*



Any triangle with angles A, B, C. A line through C parallel to the base AB returns copies of A and B at C by alternate angles, and the three angles in a row there make a straight 180°.

The triangle is the simplest shape with a true inside, and the strongest: bridges, roofs and pylons are braced with triangles because, unlike a square, a triangle cannot be pushed out of shape. It is also the workhorse of geometry –

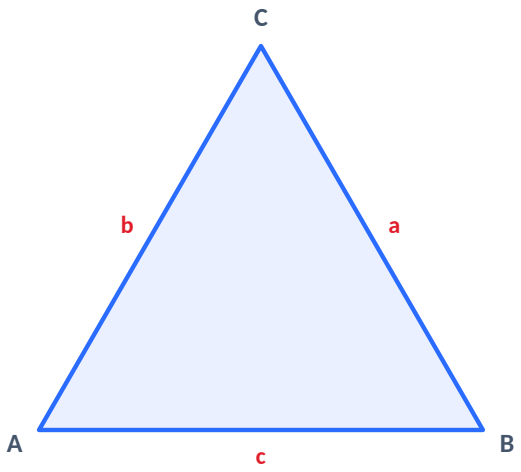
split any straight-sided shape into triangles and you can measure it. This chapter gives you the four facts about triangles that you will use forever.

Naming triangles

A **triangle** has three sides, three corners (**vertices**) and three angles. We sort them in two ways – by their **sides** and by their **angles**.

By sides: - **Equilateral** – all three sides equal (and all angles 60°). - **Isosceles** – two sides equal (the angles facing them are equal too). - **Scalene** – all three sides different.

By angles: - **Acute** – all three angles less than 90°. - **Right** – one angle exactly 90° (marked with a small square). - **Obtuse** – one angle greater than 90°.



An equilateral triangle: all three sides equal, and each angle is 60°. Sides are labelled by the vertex they start from.

> **TRIANGLE**
>
> A **triangle** is a closed shape with three straight sides and three angles. Its corners are **vertices**; the three interior angles always sum to 180°.

The angle sum: always 180°

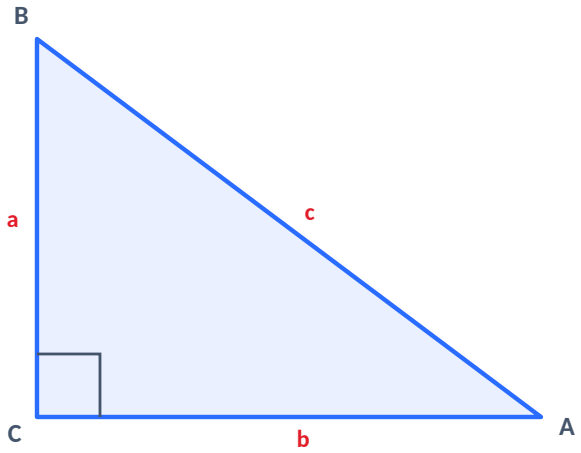
Here is the fact that runs through all of triangle geometry: **the three interior angles of any triangle add up to 180°.** Big or small, fat or thin, the three corners together make a straight angle. So if you know two angles, a single subtraction gives the third.

$$\angle A + \angle B + \angle C = 180^\circ$$

Pythagoras: the right-triangle rule

A **right triangle** has one 90° angle. The side opposite that right angle is the longest, called the **hypotenuse**. The most famous rule in mathematics links the three sides: the square of the hypotenuse equals the sum of the squares of the other two sides.

$$a^2 + b^2 = c^2 \quad (\text{where } c \text{ is the hypotenuse})$$



A right triangle with the right angle at C (small square). The hypotenuse c faces the right angle. Here legs 3 and 4 give c = 5.

HYPOTENUSE

In a **right triangle**, the **hypotenuse** is the side opposite the right angle. It is always the **longest** side, written c in $a^2 + b^2 = c^2$.

Perimeter and area

The **perimeter** is just the distance round the outside – add the three sides. The **area** (the space inside) has a tidy formula using any side as the **base** and the straight-line height to the opposite vertex.

$$\text{perimeter} = a + b + c \qquad \text{area} = \tfrac{1}{2}\,(\text{base}) \times (\text{height})$$

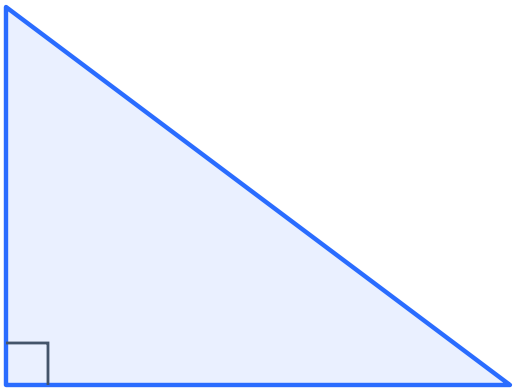
When you do not know the height but you do know all three sides, **Heron's formula** finds the area. First take the **semi-perimeter** $s = \frac{a+b+c}{2}$ (half the perimeter), then:

$$\text{area} = \sqrt{s\,(s-a)(s-b)(s-c)}$$

triangle hypotenuse semi-perimeter congruent similar

Congruent and similar triangles

Two triangles are **congruent** if they are *identical* – same side lengths and same angles, so one could be slid or flipped exactly onto the other. Two triangles are **similar** if they are the *same shape but a different size*: their angles match and their sides are all scaled by the same factor (like a photo and its enlargement).



A larger right triangle similar to the 3-4-5 one above: every side is 1.5× longer (4.5, 6, 7.5), but the angles are unchanged.

Worked examples

Example 1

Two angles of a triangle are 50° and 60°. Find the third.

SOLUTION

1. The three angles add to 180°. angles = 180°
2. Add the two known ones: $50^\circ + 60^\circ = 110^\circ$.
3. Subtract from 180°: $180^\circ - 110^\circ = 70^\circ$.

The third angle is 70°.

Example 2

Classify a triangle whose angles are 90°, 45° and 45°.

SOLUTION

1. One angle is exactly 90°, so it is a **right** triangle. 90° → right
2. Two angles are equal (45° each), so the two sides facing them are equal – it is also **isosceles**.

A right isosceles triangle.

Example 3

A right triangle has legs 3 and 4. Find the hypotenuse.

SOLUTION

1. Use $a^2 + b^2 = c^2$ with $a = 3, b = 4$. $a^2+b^2=c^2$
2. Square the legs: $3^2 + 4^2 = 9 + 16 = 25$.
3. So $c^2 = 25$, hence $c = \sqrt{25} = 5$. take $\sqrt{}$

The hypotenuse is 5. (This is the famous 3-4-5 triangle.)

Example 4

A right triangle has hypotenuse 13 and one leg 5. Find the other leg.

SOLUTION

1. Rearrange $a^2 + b^2 = c^2$ to find a leg: $b^2 = c^2 - a^2$. move a^2 over
2. Substitute $c = 13, a = 5$: $b^2 = 13^2 - 5^2 = 169 - 25 = 144$.
3. So $b = \sqrt{144} = 12$.

The other leg is 12 (a 5-12-13 triangle).

Example 5

Find the area of a triangle with base 10 and height 6.

SOLUTION

1. Use area $= \frac{1}{2} \times \text{base} \times \text{height}$. ÷2 for area
2. Substitute: $\frac{1}{2} \times 10 \times 6$.
3. Compute: $\frac{1}{2} \times 60 = 30$.

The area is 30 square units.

Example 6

Use Heron's formula to find the area of a triangle with sides $5, 6, 7$.

SOLUTION

1. Semi-perimeter: $s = \frac{5+6+7}{2} = \frac{18}{2} = 9$. half perimeter
2. Form the products: $s - a = 4$, $s - b = 3$, $s - c = 2$.
3. Plug in: area $= \sqrt{9 \times 4 \times 3 \times 2} = \sqrt{216}$.
4. Simplify: $\sqrt{216} \approx 14.7$.

The area is $\sqrt{216} \approx 14.7$ square units.

Example 7

Triangle PQR is similar to triangle ABC. In ABC the sides are $3, 4, 5$; in PQR the side matching the "3" is 9. Find the other two sides of PQR.

SOLUTION

1. Similar means every side is scaled by the same factor. The "3" became 9, so the scale factor is $9 \div 3 = 3$. *same scale*
2. Scale the matching "4": $4 \times 3 = 12$.
3. Scale the matching "5": $5 \times 3 = 15$.

PQR has sides $9, 12, 15$ (the same shape, three times as big).

Example 8

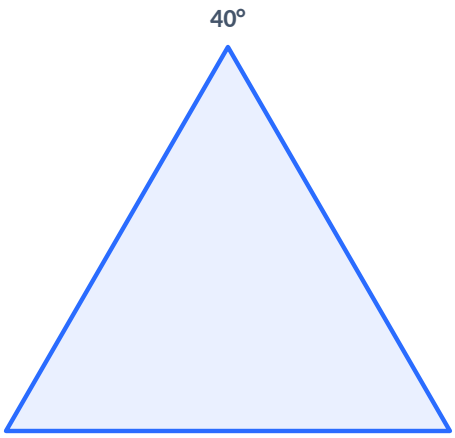
An **isosceles** triangle has an apex angle of 40°. Its two base angles are equal. Find each base angle.

SOLUTION

1. All three angles add to 180°, so the two base angles share what is left after the apex: $180^\circ - 40^\circ = 140^\circ$. *angles = 180°*
2. The base angles are equal, so split the 140° evenly between them: $140^\circ \div 2 = 70^\circ$.

isosceles → equal base angles

Each base angle is 70° (check: $40 + 70 + 70 = 180$).



An isosceles triangle with a 40° apex. The two equal sides force the two base angles to be equal, so each is (180−40)/2 = 70°.

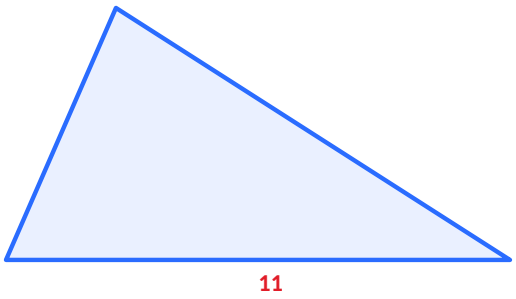
Example 9

A triangle has sides 6, 8 and 11. Is it a right triangle? (Use the converse of Pythagoras.)

SOLUTION

1. The **converse** says: if $a^2 + b^2 = c^2$ for the longest side c, the triangle is right-angled; otherwise it is not. Here $c = 11$ is the longest. *test longest as c*
2. Square the two shorter sides and add: $6^2 + 8^2 = 36 + 64 = 100$.
3. Square the longest side: $11^2 = 121$.
4. Compare: $100 \neq 121$, so the rule fails. *100 ≠ 121*

No – it is not a right triangle. (Because $121 > 100$, the angle facing the 11 is actually a touch wider than 90°, making it obtuse.)



A 6-8-11 triangle. The two short sides give $6^2+8^2 = 100$, but the long side gives $11^2 = 121$. Since $100 \neq 121$, it is not right-angled (the corner opposite 11 is obtuse).

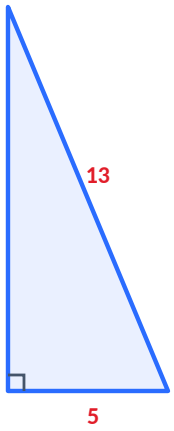
Example 10

A 13 m ladder leans against a vertical wall with its foot 5 m out from the base. How high up the wall does it reach?

SOLUTION

1. The ground, the wall and the ladder form a **right triangle**: the ladder is the hypotenuse 13, the foot-distance 5 is one leg, and the height h up the wall is the other leg. *wall ⊥ ground*
2. Rearrange Pythagoras for the unknown leg: $h^2 = c^2 - a^2 = 13^2 - 5^2 = 169 - 25 = 144$. *move 5^2 over*
3. Take the square root: $h = \sqrt{144} = 12$.

The ladder reaches 12 m up the wall (a 5-12-13 right triangle).



The ladder problem: the wall (left, height 12) and ground (bottom, 5 m out) meet at a right angle, and the 13 m ladder is the hypotenuse. Pythagoras gives the height 12.

Note: Pythagoras works **only** in right triangles, and c must be the hypotenuse (the side facing the right angle). If you square the wrong side as c, the answer will be off. Always check which angle is the 90° one first.

Mental-Math Shortcut

Memorise a few **Pythagorean triples** – whole-number right triangles – and you can read off a hypotenuse instantly: **3-4-5**, **5-12-13**, **8-15-17**, and **7-24-25**. Better still, every triple can be *scaled*: doubling 3-4-5 gives **6-8-10**, tripling gives **9-12-15**, and so on. So if you spot legs of 6 and 8, you do not need to square anything – it is just a 3-4-5 scaled by 2, so the hypotenuse is 10. Recognising the pattern beats the arithmetic every time.

Why do a triangle's angles **always** total 180°, for *every* triangle? The picture-proof uses parallel lines and the angle facts from the last chapter, and it is a lovely chain of "therefore"s.

Take any triangle with angles A, B, C. Through the top vertex draw a line **parallel** to the base. The base now acts as a transversal cutting two parallel lines, and so does each of the other two sides.

By the **alternate angles** rule (the "Z"-shape, equal between parallels), the angle the left side makes with the new parallel line equals A, and the angle the right side makes with it equals B. So along that straight parallel line, sitting side by side, we now have three angles: a copy of A, then C at the vertex, then a copy of B.

But angles in a row on a **straight line** add to 180°. Therefore

$$A + C + B = 180^\circ.$$

The reasoning never used the triangle's size or shape – only "alternate angles are equal" and "a straight line is 180°". Because it assumed *nothing special*, it holds for **all** triangles. That is the difference between checking one example and **proving** a universal truth.

⚠ Common Mistakes

- Using Pythagoras on a triangle with no right angle. $a^2 + b^2 = c^2$ holds **only** for right triangles.
- Putting a leg where the hypotenuse goes. In $a^2 + b^2 = c^2$, c must be the side **opposite** the right angle (the longest one).
- Forgetting the $\frac{1}{2}$ in the area formula. Area is $\frac{1}{2} \times \text{base} \times \text{height}$, not base × height.
- Using a slanted side as the "height". The height must be the **perpendicular** distance from the base to the opposite vertex.
- Thinking similar triangles have equal sides. Similar means equal **angles** and **proportional** sides; only *congruent* triangles have equal sides.

Quiz

46.1.Q1. The three interior angles of any triangle add to ___.

A. 90°

B. 180°

C. 270°

D. 360°

46.1.Q2. Two angles of a triangle are 70° and 55°. Find the third.

46.1.Q3. A triangle with all three sides equal is called ___.

A. Scalene

B. Isosceles

C. Equilateral

D. Right

46.1.Q4. A right triangle has legs 6 and 8. Find the hypotenuse.

A. 10

B. 12

C. 14

D. 48

46.1.Q5. In a right triangle, which side is the hypotenuse?

A. The shortest

B. Any leg

C. The side opposite the right angle

D. The base

46.1.Q6. Find the area of a triangle with base 12 and height 5.

46.1.Q7. True or False: Pythagoras' rule works for every triangle.

A. True

B. False

46.1.Q8. Triangle A has sides $2, 3, 4$; similar triangle B has the matching "2" equal to 6. What is its longest side?

46.1.Q9. What is the semi-perimeter s in Heron's formula for sides $5, 6, 7$?

🐍 From Concept to Code (Python)

Two of this chapter's formulas turn straight into short functions. The first uses Pythagoras to get a hypotenuse from two legs; the second uses Heron's formula to get the area from all three sides. Both need square roots, so we import the standard `math` module.

```
import math

def hypotenuse(a, b):
    return math.sqrt(a*a + b*b)          # Pythagoras

def heron(a, b, c):
    s = (a + b + c) / 2                  # semi-perimeter
    return math.sqrt(s*(s-a)*(s-b)*(s-c))

print(hypotenuse(3, 4))                  # the 3-4-5 triangle
print(round(heron(5, 6, 7), 2))          # area from three sides
```

```
output
5.0
14.7
```

Try it:

1. Use `hypotenuse` to confirm a 5-12-13 triangle: `hypotenuse(5, 12)` should print `13.0`.
2. Compute `heron(3, 4, 5)`. It should give `6.0` – check it matches $\frac{1}{2} \times 3 \times 4$ for this right triangle.

Answer Key – Chapter 46 – Triangles

46.1.Q1. 180° – true for every triangle.

46.1.Q2. 55° – $180 - (70 + 55) = 180 - 125 = 55$.

46.1.Q3. Equilateral – all sides (and all angles, 60°) equal.

46.1.Q4. 10 – $6^2 + 8^2 = 36 + 64 = 100$, and $\sqrt{100} = 10$.

46.1.Q5. The side opposite the right angle – always the longest.

46.1.Q6. 30 – $\frac{1}{2} \times 12 \times 5 = 30$ square units.

46.1.Q7. False – only for **right** triangles.

46.1.Q8. 12 – the scale factor is $6 \div 2 = 3$, so $4 \times 3 = 12$.

46.1.Q9. 9 – $s = (5 + 6 + 7)/2 = 18/2 = 9$.

Chapter 47 · Circles

The parts of a circle and the number π that ties them together, how to find circumference and area from the radius, how arcs and sectors take a slice of each, and the difference between a chord and a tangent.

The Idea Behind It

Circumference ÷ diameter is the same number π for *every* circle, a coin or the Moon's orbit.

but why?

Measure the way round a circle and divide by the way across, and you always land just past 3, whatever the circle's size. Why should a tiny circle and a huge one agree? Because all circles are the same shape, and that one fact does the work:

$$\boxed{\text{all circles are similar: enlarging by a factor } k \text{ scales every length by } k}$$

Take a small circle and a big one. Being the same shape, the big one is just the small one enlarged by some factor k, so its circumference **and** its diameter are each k times the small circle's. Now divide, and the k on top cancels the k on the bottom:

$$\frac{\text{big } C}{\text{big } d} = \frac{k \cdot C_{\text{small}}}{k \cdot d_{\text{small}}} = \frac{C_{\text{small}}}{d_{\text{small}}}$$

Both circles give the *same* ratio, and the two circles were arbitrary, so **every** circle gives that one shared value. We name it π:

$$\therefore \quad \frac{C}{d} = \pi \approx 3.14159\ldots \text{ for all circles}$$

So π is not a fact about any one circle but about *circle-ness* itself, which is why a single constant serves them all.

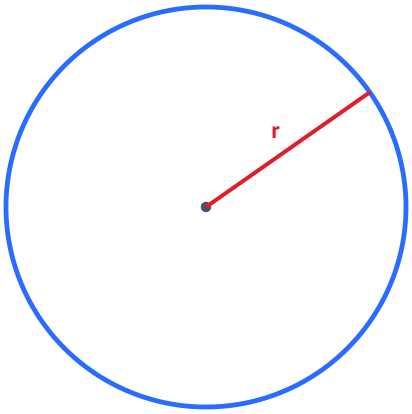
cancelling the scale factor turns "I checked a few" into "it holds for all"

A wheel, a coin, a clock face, a ripple on a pond, the orbit of a planet: the circle is everywhere because it is the shape of perfect fairness – every point on it is the *same distance* from the middle. That one property gives the circle all its formulas, and a single special number, π, unlocks them. This chapter shows where π comes from and how to measure any circle from just its radius.

The parts of a circle

A **circle** is the set of all points a fixed distance from a centre. That fixed distance is the **radius** (r). A straight line from one side to the other *through* the centre is the **diameter** (d), and it is exactly twice the radius. The distance once around the outside is the **circumference**.

$$d = 2r \qquad \Longleftrightarrow \qquad r = \tfrac{d}{2}$$



A circle with centre marked. The red segment is a radius r; a full line across through the centre would be the diameter d = 2r.

CIRCLE

A **circle** is all the points a fixed distance – the **radius** r – from a centre point. The **diameter** $d = 2r$ crosses through the centre; the **circumference** is the distance around it.

The number π

Measure the way round any circle (its circumference) and divide by the way across (its diameter). You always get the *same* number, a little more than 3, no matter the circle's size. That ratio is π (pi):

$$\pi = \frac{\text{circumference}}{\text{diameter}} \approx 3.14159\ldots$$

π never ends and never repeats, but for most work $\pi \approx 3.14$ (or the fraction $\frac{22}{7}$) is plenty.

PI (Π)

π is the constant ratio of any circle's circumference to its diameter, about 3.14159. It appears in every circle formula.

Circumference and area

Because π is the circumference divided by the diameter, multiplying back gives the **circumference**. The **area** (the space inside) uses the radius squared.

$$\text{circumference } C = 2\pi r = \pi d \qquad\qquad \text{area } A = \pi r^2$$

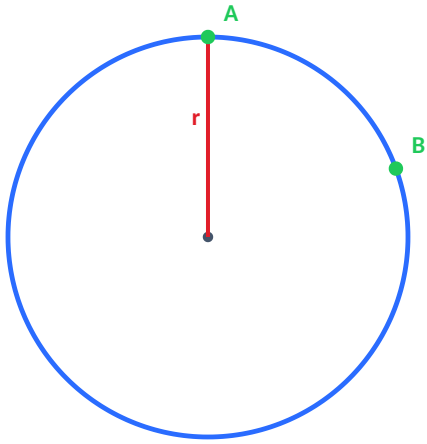
Tip: Tell the two apart by the units. Circumference is a length, so it uses r to the first power ($2\pi r$). Area covers a surface, so it uses r **squared** (πr^2) – "area has the square".

radius diameter circumference chord tangent sector

Arcs and sectors: slices of the circle

Take a *fraction* of the journey round the edge and you have an **arc** – part of the circumference. Take the matching *wedge* of the inside (two radii and that arc, like a pizza slice) and you have a **sector** – part of the area. To find either, multiply the whole-circle formula by the fraction of the full 360° that your angle covers.

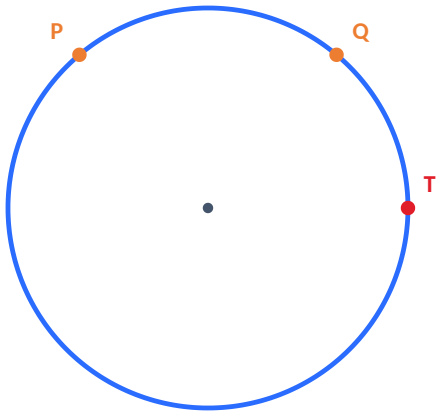
$$\text{arc length} = \frac{\theta}{360^\circ} \times 2\pi r \qquad \text{sector area} = \frac{\theta}{360^\circ} \times \pi r^2$$



A sector: the wedge between two radii (to A and B) and the arc joining them. Its share of the circle equals its angle ÷ 360°.

Chords and tangents

Two more lines matter. A **chord** is a straight line joining two points *on* the circle; the longest possible chord is the diameter. A **tangent** is a line that just grazes the circle, touching at a *single* point – and at that point the tangent is always at right angles (90°) to the radius. (A car tyre meets the road along a tangent.)



The orange line PQ is a chord (joins two points on the circle). At T a tangent would touch the circle once, meeting the radius to T at 90°.

Worked examples

Example 1

A circle has radius 7. Find its diameter and circumference (use $\pi \approx 3.14$).

SOLUTION

1. Diameter is twice the radius: $d = 2 \times 7 = 14$. *d = 2r*
2. Circumference $C = 2\pi r = 2 \times 3.14 \times 7$.
3. Compute: $2 \times 3.14 = 6.28$, then $6.28 \times 7 = 43.96$.

Diameter 14; circumference ≈ 43.96 units.

Example 2

Find the area of a circle of radius 5 (use $\pi \approx 3.14$).

SOLUTION

1. Area $A = \pi r^2$, so square the radius first: $5^2 = 25$. *square r first*
2. Multiply by π: 3.14×25.
3. Compute: $3.14 \times 25 = 78.5$.

The area is ≈ 78.5 square units.

Example 3

A circle has diameter 20. Find its radius and area (use $\pi \approx 3.14$).

SOLUTION

1. Radius is half the diameter: $r = 20 \div 2 = 10$. *r = d/2*
2. Area $= \pi r^2 = 3.14 \times 10^2 = 3.14 \times 100$.
3. Compute: $3.14 \times 100 = 314$.

Radius 10; area ≈ 314 square units.

Example 4

A circle has circumference ≈ 31.4. Find its radius (use $\pi \approx 3.14$).

SOLUTION

1. Start from $C = 2\pi r$ and solve for r: $r = \dfrac{C}{2\pi}$. *÷ both by 2π*
2. Substitute: $r = \dfrac{31.4}{2 \times 3.14} = \dfrac{31.4}{6.28}$.
3. Divide: $31.4 \div 6.28 = 5$.

The radius is 5.

Example 5

Find the length of a 90° arc on a circle of radius 6 (use $\pi \approx 3.14$).

SOLUTION

1. A 90° arc is $\frac{90}{360} = \frac{1}{4}$ of the full circumference. slice = θ/360
2. Full circumference: $2\pi r = 2 \times 3.14 \times 6 = 37.68$.
3. Take a quarter: $37.68 \div 4 = 9.42$.

The arc length is ≈ 9.42 units.

Example 6

Find the area of a 90° sector of a circle of radius 6 (use $\pi \approx 3.14$).

SOLUTION

1. A 90° sector is $\frac{90}{360} = \frac{1}{4}$ of the whole area. quarter slice
2. Whole area: $\pi r^2 = 3.14 \times 36 = 113.04$.
3. Take a quarter: $113.04 \div 4 = 28.26$.

The sector area is ≈ 28.26 square units.

Example 7

A tangent touches a circle at point T. The radius to T is 4 and a point on the tangent is 3 away from T. How far is that point from the centre?

SOLUTION

1. The tangent meets the radius at T at a right angle, so we have a **right triangle**: legs 4 (radius) and 3 (along the tangent). tangent ⊥ radius
2. Use Pythagoras: distance$^2 = 4^2 + 3^2 = 16 + 9 = 25$. $a^2+b^2=c^2$
3. So the distance $= \sqrt{25} = 5$.

The point is 5 from the centre (a tidy 3-4-5 right triangle).

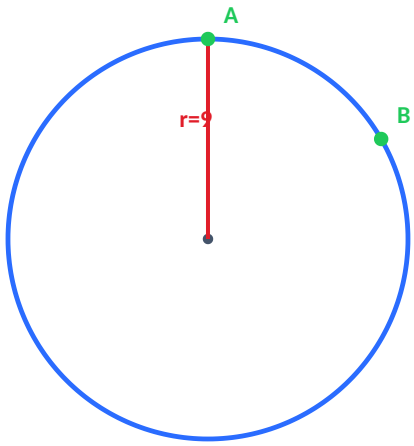
Example 8

Find the area of a 60° sector of a circle of radius 9 (use $\pi \approx 3.14$).

SOLUTION

1. A 60° sector is $\frac{60}{360} = \frac{1}{6}$ of the whole circle. *slice = θ/360*
2. Whole area: $\pi r^2 = 3.14 \times 9^2 = 3.14 \times 81 = 254.34$ square units.
3. Take one sixth: $254.34 \div 6 = 42.39$.

The sector area is ≈ 42.39 square units (a sixth of the circle, because 60° is a sixth of 360°).



A 60° sector (the wedge between the two radii to A and B). Its 60° opening is one sixth of the full 360°, so it holds one sixth of the area.

Example 9

A circle has area ≈ 50.24. Find its radius (use $\pi \approx 3.14$).

SOLUTION

1. Start from $A = \pi r^2$ and solve for r^2 by dividing by π: $r^2 = \frac{A}{\pi} = \frac{50.24}{3.14} = 16$. *÷ both by π*
2. Undo the square by taking the square root: $r = \sqrt{16} = 4$. *take √, not ÷2*

The radius is 4. (Going backwards from area needs a square root, where going backwards from circumference only needed a division.)

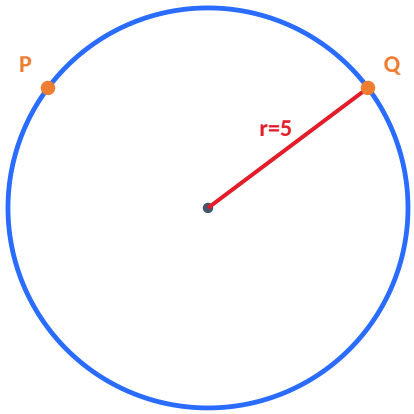
Example 10

A chord is 8 long in a circle of radius 5. The perpendicular from the centre cuts the chord exactly in half. How far is the chord from the centre?

SOLUTION

1. The perpendicular from the centre **bisects** the chord, so each half is $8 \div 2 = 4$.
   ⊥ *from centre bisects chord*
2. This makes a **right triangle**: the radius 5 to a chord-end is the hypotenuse, the half-chord 4 is one leg, and the distance d from centre to chord is the other leg. *radius = hypotenuse*
3. Pythagoras for the leg: $d^2 = 5^2 - 4^2 = 25 - 16 = 9$, so $d = \sqrt{9} = 3$.

The chord is 3 from the centre (yet another 3-4-5 right triangle hiding inside the circle).



The chord PQ has length 8; the radius to Q is 5. Dropping a perpendicular from the centre to the midpoint of PQ makes a right triangle with legs 4 (half-chord) and 3 (distance to chord).

Note: Keep the powers straight: circumference uses r once ($2\pi r$) and gives a **length**; area uses r **squared** (πr^2) and gives a **surface**. Mixing them up is the most common circle error.

Mental-Math Shortcut

To estimate a circumference in your head, use $\pi \approx 3$ and round: a circle of radius 7 has $C \approx 2 \times 3 \times 7 = 42$ (the true value 43.96 is just a touch more, because π is a touch more than 3). For area, square the radius then "times three": radius $5 \to 25 \to 75$ (true area 78.5). The rule of thumb: your "$\times 3$" answer is always slightly **under** the real one, by about 5%, because you shaved π down from 3.14 to 3. Quick, and close enough to sanity-check any calculator answer.

Why is π the **same number for every circle**, from a coin to the Moon's orbit? The reason is that all circles are **similar** – the same shape at different sizes – and similar figures scale *every* length by one common factor.

Reason it through. Take a small circle and a big one. Because they are the same shape, the big circle is just the small one enlarged by some factor k: every length is multiplied by k. In particular the big circumference is k times the small circumference, **and** the big diameter is k times the small diameter. Now divide:

$$\frac{\text{big circumference}}{\text{big diameter}} = \frac{k \cdot (\text{small circumference})}{k \cdot (\text{small diameter})} = \frac{\text{small circumference}}{\text{small diameter}}.$$

The factor k **cancels** top and bottom. So both circles give the *same* ratio – and since the two circles were arbitrary, *every* circle gives that same ratio. We give that one shared value a name: π.

This is why a single constant can serve all circles at once: it is not a property of any particular circle, but of *circle-ness* itself. The cancelling-of-the-scale-factor argument is exactly how mathematicians turn "I checked a few" into "it must hold for all".

⚠ Common Mistakes

- Using the diameter where the formula wants the radius. In $A = \pi r^2$ and $C = 2\pi r$, r is the **radius** ($= d/2$).
- Swapping the two formulas. **Circumference** $= 2\pi r$ (a length); **area** $= \pi r^2$ (uses r squared).
- Squaring the whole of $2r$ in the area: $A = \pi r^2$ squares only the radius, not the diameter.
- Forgetting to take the fraction for arcs/sectors. A 90° slice is $\frac{90}{360} = \frac{1}{4}$ of the whole, not the whole.
- Confusing a chord with a tangent. A **chord** joins two points on the circle; a **tangent** touches at just **one** point.

Quiz

47.1.Q1. The diameter of a circle is how many times its radius?

A. Half

B. Equal

C. Twice

D. Four times

47.1.Q2. What does π equal (to 2 decimal places)?

A. 2.72

B. 3.14

C. 1.62

D. 6.28

47.1.Q3. Find the circumference of a circle with radius 10 (use $\pi \approx 3.14$).

47.1.Q4. Find the area of a circle with radius 3 (use $\pi \approx 3.14$).

47.1.Q5. Which formula gives the area of a circle?

A. $2\pi r$

B. πr^2

C. πd

D. $\frac{1}{2}r$

47.1.Q6. A line that touches a circle at exactly one point is a ___.

A. Chord

B. Radius

C. Diameter

D. Tangent

47.1.Q7. A 90° sector is what fraction of the whole circle?

A. $\frac{1}{2}$

B. $\frac{1}{3}$

C. $\frac{1}{4}$

D. $\frac{1}{8}$

47.1.Q8. True or False: the longest chord of a circle is its diameter.

A. True

B. False

47.1.Q9. A circle has diameter 14. What is its radius?

🐍 From Concept to Code (Python)

Python's `math` module carries π to full precision as `math.pi`, so the two headline formulas become one-line functions. No more rounding π by hand – the computer keeps all the digits.

```
import math

def circumference(r):
    return 2 * math.pi * r        # C = 2 π r

def area(r):
    return math.pi * r**2         # A = π r²

print(round(circumference(7), 2))
print(round(area(5), 2))
```

```
output
43.98
78.54
```

Try it:

1. Write `sector_area(r, deg)` that returns `(deg/360) * math.pi * r**2`. Test a `90°` sector of radius `6` (expect about `28.27`).
2. Given a circumference, recover the radius: `r = C / (2*math.pi)`. Check that a circumference of `62.83` gives `r ≈ 10`.

Answer Key – Chapter 47 – Circles

47.1.Q1. Twice – $d = 2r$.

47.1.Q2. 3.14 – the circumference-to-diameter ratio.

47.1.Q3. 62.8 – $2\pi r = 2 \times 3.14 \times 10 = 62.8$.

47.1.Q4. 28.26 – $\pi r^2 = 3.14 \times 9 = 28.26$ square units.

47.1.Q5. πr^2 – area uses the radius squared.

47.1.Q6. Tangent – it grazes the circle once, perpendicular to the radius there.

47.1.Q7. $\frac{1}{4}$ – because $90/360 = 1/4$.

47.1.Q8. True – the diameter passes through the centre and is the longest possible chord.

47.1.Q9. 7 – the radius is half the diameter, $14 \div 2 = 7$.

Chapter 48 · Coordinate Geometry

How to turn geometry into arithmetic on the x–y grid: measuring the distance between two points, finding the exact midpoint, computing the gradient (steepness) of a line, and writing its equation y = mx + c.

The Idea Behind It

The distance between (x_1, y_1) and (x_2, y_2) is $\sqrt{(\Delta x)^2 + (\Delta y)^2}$. *but why?*

This is not a new rule to memorise: it is **Pythagoras in disguise**. Follow the picture and you never have to recall it cold. The whole derivation rests on the right-triangle rule from the last chapter:

$$\boxed{a^2 + b^2 = c^2 \quad (\text{legs } a, b;\ \text{hypotenuse } c)}$$

Drop a horizontal line from one point and a vertical line from the other. They meet at a corner, building a **right triangle** whose hypotenuse is exactly the straight-line gap d between the two points. Its legs are the run across and the rise up:

$$\text{run} = x_2 - x_1 = \Delta x, \qquad \text{rise} = y_2 - y_1 = \Delta y$$

Feed those legs into Pythagoras, with the distance d as the hypotenuse:

$$d^2 = (\Delta x)^2 + (\Delta y)^2$$

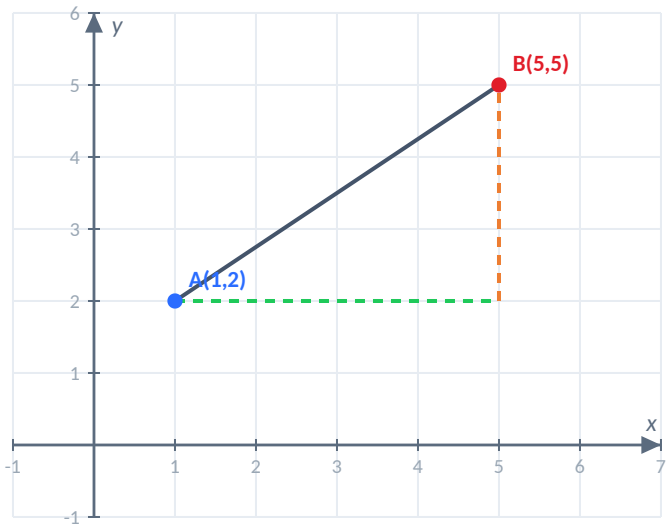
$$\therefore\ d = \sqrt{(\Delta x)^2 + (\Delta y)^2}$$

Squaring quietly handles the signs too: a negative difference squares to a positive, so the order you subtract in never changes the distance. *every grid distance is just this one right triangle in new numbers*

Geometry on the grid is geometry you can *calculate*. Once a point is a pair of numbers, the distance between two points, the steepness of a line and its whole equation all come from simple arithmetic – no protractor, no ruler. This is the bridge between the shapes of the last few chapters and the algebra of Part VI, and it is exactly how a computer "does geometry": everything becomes coordinates.

The setup: points as number pairs

Recall the **coordinate plane**: a horizontal x-axis and a vertical y-axis crossing at the origin $(0, 0)$. Every point is an ordered pair (x, y) – first across, then up. From two such points we will build everything in this chapter.



Two points A(1,2) and B(5,5). The green run (4 across) and orange rise (3 up) form a right triangle with the navy segment AB as its hypotenuse.

Rise and run

Look at the right triangle in the picture. Going from A to B you move **across** by the difference in x-values (the **run**) and **up** by the difference in y-values (the **rise**).

$$\text{run} = x_2 - x_1 \qquad\qquad \text{rise} = y_2 - y_1$$

Those two numbers, the rise and the run, are all we need for distance, gradient and the line's equation.

Distance: Pythagoras on the grid

The straight-line **distance** between two points is just the hypotenuse of that rise-run triangle. Pythagoras ($a^2 + b^2 = c^2$ from the last chapter) gives:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

DISTANCE FORMULA

The distance between (x_1, y_1) and (x_2, y_2) is $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ – Pythagoras applied to the run and the rise.

Midpoint: the average of the ends

The **midpoint** sits exactly halfway between two points. Its coordinates are simply the **averages** of the two x's and the two y's.

$$\text{midpoint} = \left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2} \right)$$

Gradient: how steep the line is

The **gradient** (or **slope**), written m, measures steepness as **rise over run** – how much the line climbs for each step across. A positive gradient rises left-to-right; a negative one falls; zero is flat.

$$m = \frac{\text{rise}}{\text{run}} = \frac{y_2 - y_1}{x_2 - x_1}$$

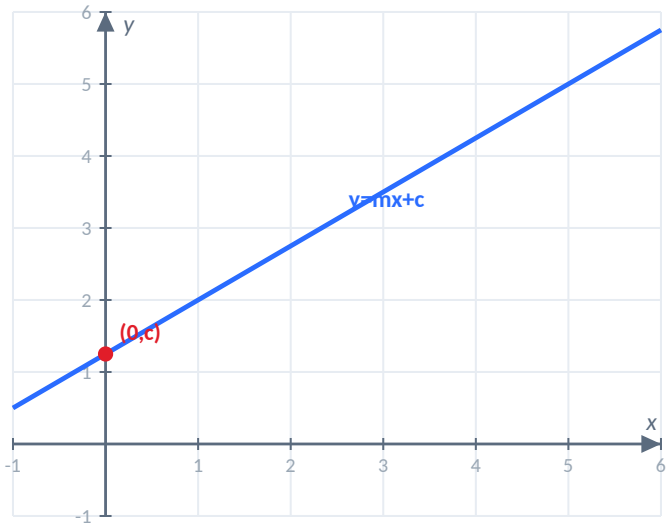
> **GRADIENT**
>
> The **gradient** (slope) m of a line is its **rise over run**: $m = \dfrac{y_2 - y_1}{x_2 - x_1}$. It is the amount the line climbs for each 1 step to the right.

distance formula midpoint gradient y-intercept

The equation of a line: y = mx + c

Every straight line can be written $y = mx + c$, where m is the **gradient** and c is the **y-intercept** – the height at which the line crosses the y-axis. Knowing m and c lets you draw the line or find the y for any x.

$$y = mx + c \qquad (m = \text{gradient},\ c = y\text{-intercept})$$



A line y = mx + c. It crosses the y-axis at the intercept (0, c); the gradient m sets how steeply it rises.

Two lines are **parallel** when they have the **same gradient** (they rise at the same rate, never meeting). Two lines are **perpendicular** (meeting at 90°) when their gradients multiply to -1 – each is the *negative reciprocal* of the other (e.g. 2 and $-\frac{1}{2}$).

Worked examples

Example 1

Find the distance between $A(1, 2)$ and $B(5, 5)$.

SOLUTION

1. Run $= x_2 - x_1 = 5 - 1 = 4$; rise $= y_2 - y_1 = 5 - 2 = 3$. rise & run
2. Square and add: $4^2 + 3^2 = 16 + 9 = 25$. $a^2+b^2=c^2$
3. Take the square root: $\sqrt{25} = 5$.

The distance is 5 (the 3-4-5 triangle in the picture above).

Example 2

Find the midpoint of $A(1, 2)$ and $B(5, 5)$.

SOLUTION

1. Average the x-values: $\dfrac{1+5}{2} = \dfrac{6}{2} = 3$. average the ends
2. Average the y-values: $\dfrac{2+5}{2} = \dfrac{7}{2} = 3.5$.

The midpoint is $(3,\ 3.5)$.

Example 3

Find the gradient of the line through $A(1, 2)$ and $B(5, 5)$.

SOLUTION

1. Gradient $m = \dfrac{\text{rise}}{\text{run}} = \dfrac{y_2 - y_1}{x_2 - x_1}$. rise/run
2. Substitute: $m = \dfrac{5-2}{5-1} = \dfrac{3}{4}$.

The gradient is $\frac{3}{4}$ (it climbs 3 for every 4 across).

Example 4

A line has gradient 2 and crosses the y-axis at $(0, -1)$. Write its equation, then find y when $x = 3$.

SOLUTION

1. Here $m = 2$ and the y-intercept $c = -1$, so $y = 2x - 1$. y = mx + c
2. Substitute $x = 3$: $y = 2(3) - 1 = 6 - 1 = 5$.

The line is $y = 2x - 1$, and at $x = 3$, $y = 5$.

Example 5

Find the equation of the line through $A(1, 2)$ and $B(5, 5)$.

SOLUTION

1. From Example 3 the gradient is $m = \frac{3}{4}$, so $y = \frac{3}{4}x + c$.
2. Find c by substituting one point, say $A(1, 2)$: $2 = \frac{3}{4}(1) + c$. sub in a point
3. Solve: $c = 2 - \frac{3}{4} = \frac{5}{4}$.

The line is $y = \frac{3}{4}x + \frac{5}{4}$.

Example 6

Is the line through $(0, 1)$ and $(2, 5)$ parallel to $y = 2x + 7$?

SOLUTION

1. Find the first line's gradient: $m = \frac{5 - 1}{2 - 0} = \frac{4}{2} = 2$. rise/run
2. The line $y = 2x + 7$ has gradient 2.
3. Equal gradients (and different intercepts) means the lines never meet. slopes match

Yes – both have gradient 2, so they are parallel.

Example 7

A line has gradient 3. What gradient is perpendicular to it?

SOLUTION

1. Perpendicular gradients multiply to -1: $3 \times m = -1$. product = -1
2. Solve for m: $m = -\frac{1}{3}$ (the negative reciprocal of 3).

A perpendicular line has gradient $-\frac{1}{3}$.

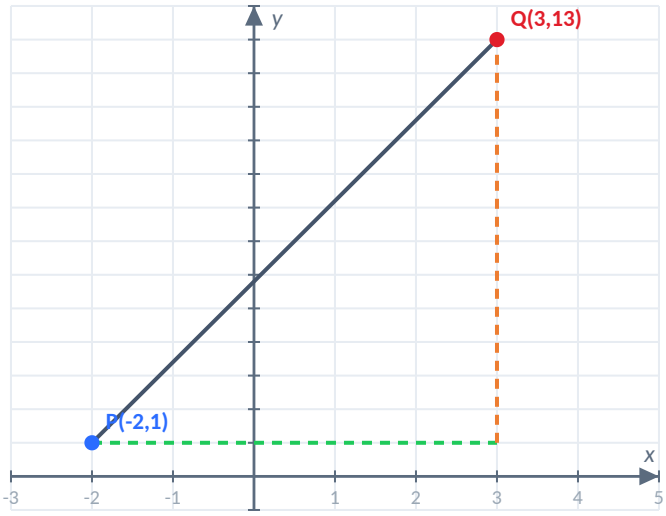
Example 8

Find the distance between $P(-2,\ 1)$ and $Q(3,\ 13)$.

SOLUTION

1. Run $= x_2 - x_1 = 3 - (-2) = 5$; rise $= y_2 - y_1 = 13 - 1 = 12$. mind the -(-2)
2. Square and add: $5^2 + 12^2 = 25 + 144 = 169$. $a^2+b^2=c^2$
3. Take the square root: $\sqrt{169} = 13$.

The distance is 13 – a 5-12-13 right triangle living on the grid.



From P to Q the run is 5 (green) and the rise is 12 (orange). The navy hypotenuse PQ has length $\sqrt{(5^2+12^2)} = 13$.

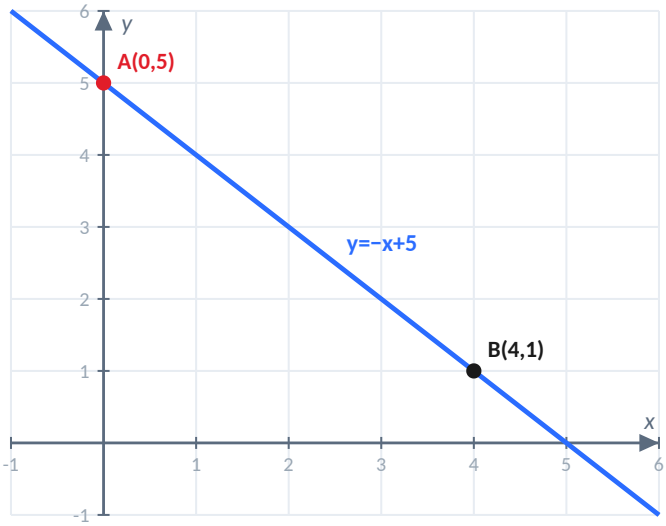
Example 9

Find the equation of the line through $A(0,\ 5)$ and $B(4,\ 1)$.

SOLUTION

1. Gradient first: $m = \dfrac{y_2 - y_1}{x_2 - x_1} = \dfrac{1-5}{4-0} = \dfrac{-4}{4} = -1$. falls → negative slope
2. The line meets the y-axis at $A(0, 5)$, so the intercept is read straight off: $c = 5$. x=0 gives c
3. Put m and c into $y = mx + c$.

The line is $y = -x + 5$ (it falls one step for every step across, crossing the y-axis at 5).



The line through A(0,5) and B(4,1). Its gradient is −1 (it falls), and it cuts the y-axis at 5, giving y = −x + 5.

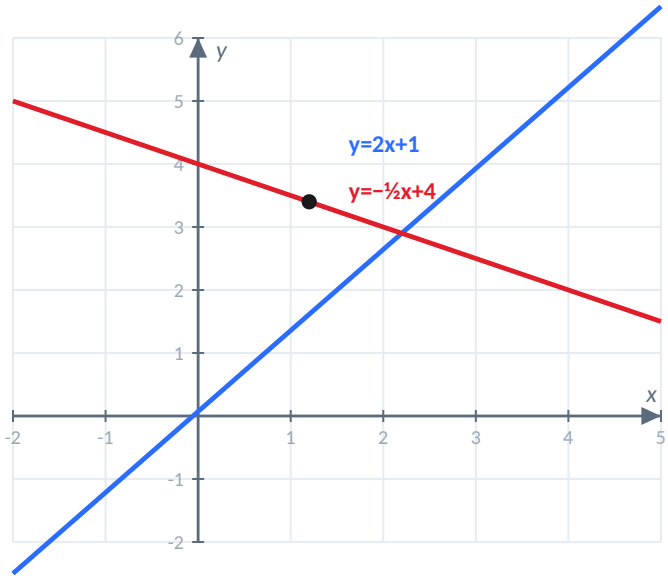
Example 10

Are the lines $y = 2x + 1$ and $y = -\frac{1}{2}x + 4$ perpendicular?

SOLUTION

1. Read off the two gradients: $m_1 = 2$ and $m_2 = -\frac{1}{2}$. gradients are the x-coefficients
2. Perpendicular lines have gradients whose **product is** -1. Multiply: $2 \times \left(-\frac{1}{2}\right) = -1$.
3. The product is exactly -1, so the test passes. product = −1 ✓

Yes – the lines are perpendicular; they cross at a right angle (90°).



Two lines whose gradients (2 and −½) multiply to −1. They meet at a right angle, so they are perpendicular.

Note: Order does not matter in distance or midpoint (you add or average the two points), but it **must be consistent** in the gradient: if $y_2 - y_1$ is on top, then $x_2 - x_1$ must be on the bottom – subtract the points in the **same order** both times, or the sign of the slope flips.

Mental-Math Shortcut

For the **midpoint**, average each coordinate by the schoolyard trick: add the two numbers and halve. Even easier when they are far apart – find the *gap*, halve it, and step that far from the smaller one. Midpoint of $x = 4$ and $x = 18$: the gap is 14, half is 7, so $4 + 7 = 11$. For a quick **distance** estimate, recognise Pythagorean triples on the grid: a run of 6 and rise of 8 is a scaled 3-4-5, so the distance is 10 – no squaring needed. Spotting 3-4-5, 6-8-10 and 5-12-13 in the rise and run turns many distance questions into instant answers.

Where does the **distance formula** come from? It is not a new rule to memorise – it is **Pythagoras in disguise**, and following the logic means you never have to recall it cold.

Take two points (x_1, y_1) and (x_2, y_2). Drop a horizontal line from one and a vertical line from the other; they meet at a corner, forming a **right triangle** whose hypotenuse is the segment joining the two points (this is exactly the dashed triangle in the opening figure). Now reason in three steps:

1. The horizontal leg (the run) has length $x_2 - x_1$ – the change in the across-direction.
2. The vertical leg (the rise) has length $y_2 - y_1$ – the change in the up-direction.
3. By Pythagoras, hypotenuse^2 = leg^2 + leg^2, so $d^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2$.

Take the square root of both sides and the distance formula falls out. (The squaring quietly handles the signs, too – even if a difference comes out negative, its square is positive, so the order you subtract in does not change the distance.) Every coordinate-geometry distance you will ever compute is just this one right triangle, dressed in different numbers.

⚠ Common Mistakes

- Subtracting the points in a different order on top and bottom of the gradient. Keep it consistent: $\dfrac{y_2 - y_1}{x_2 - x_1}$, same order both times.
- Forgetting the square root in the distance formula. d^2 is the sum of squares; the distance d is its **square root**.
- Subtracting instead of averaging for the midpoint. The midpoint **averages** the coordinates: $\left(\dfrac{x_1 + x_2}{2}, \dfrac{y_1 + y_2}{2}\right)$.
- Reading c as the gradient. In $y = mx + c$, m is the gradient and c is the y**-intercept**.
- Thinking perpendicular lines have the same gradient. **Parallel** lines share a gradient; **perpendicular** gradients multiply to -1.

Quiz

48.1.Q1. Find the distance between $(0, 0)$ and $(3, 4)$.

A. 5

B. 7

C. 12

D. 25

48.1.Q2. Find the midpoint of $(2, 4)$ and $(6, 10)$.

48.1.Q3. Find the gradient of the line through $(1, 1)$ and $(3, 7)$.

A. 3

B. 2

C. $\frac{1}{3}$

D. 6

48.1.Q4. In $y = mx + c$, which letter is the y-intercept?

A. m

B. x

C. c

D. y

48.1.Q5. A line is $y = 4x - 3$. What is y when $x = 2$?

48.1.Q6. Two lines both have gradient 5. They are ___.

A. Perpendicular

B. Parallel

C. The same line

D. Intersecting at 90°

48.1.Q7. What gradient is perpendicular to a line of gradient 2?

A. 2

B. -2

C. $\frac{1}{2}$

D. $-\frac{1}{2}$

48.1.Q8. True or False: the distance between two points can be negative.

A. True

B. False

48.1.Q9. A line has gradient $\frac{1}{2}$ and passes through $(0, 3)$. Write its equation.

🐍 From Concept to Code (Python)

The distance and midpoint formulas become two short functions that each take two points (as tuples). Distance needs a square root, so we import `math`; the midpoint is pure averaging.

```
import math

def distance(p, q):
    return math.sqrt((q[0]-p[0])**2 + (q[1]-p[1])**2)

def midpoint(p, q):
    return ((p[0]+q[0])/2, (p[1]+q[1])/2)

A, B = (1, 2), (5, 5)
print(distance(A, B))     # the 3-4-5 hypotenuse
print(midpoint(A, B))
```

```
output
5.0
(3.0, 3.5)
```

Try it:

1. Use `distance` on `(0,0)` and `(6,8)`. Confirm it prints `10.0` (a scaled 3-4-5 triangle).
2. Write `gradient(p, q)` returning `(q[1]-p[1]) / (q[0]-p[0])` and test it on `(1,1)` and `(3,7)` (expect `3.0`).

Answer Key – Chapter 48 – Coordinate Geometry

48.1.Q1. 5 – $\sqrt{3^2 + 4^2} = \sqrt{25} = 5$.

48.1.Q2. $(4, 7)$ – average each coordinate: $(2 + 6)/2 = 4$ and $(4 + 10)/2 = 7$.

48.1.Q3. 3 – rise over run $= \dfrac{7 - 1}{3 - 1} = \dfrac{6}{2} = 3$.

48.1.Q4. c – the height where the line crosses the y-axis; m is the gradient.

48.1.Q5. 5 – $4(2) - 3 = 8 - 3 = 5$.

48.1.Q6. Parallel – equal gradients means they rise at the same rate and never meet.

48.1.Q7. $-\frac{1}{2}$ – the negative reciprocal, since $2 \times (-\frac{1}{2}) = -1$.

48.1.Q8. False – a distance is a length, always ≥ 0; the square root is taken as positive.

48.1.Q9. $y = \frac{1}{2}x + 3$ – gradient $m = \frac{1}{2}$, intercept $c = 3$.

Chapter 49 · Distance, Area & Volume

How to measure shapes: the distance around a flat shape (perimeter), the space inside it (area), the skin of a solid (surface area) and the space a solid fills (volume) – for rectangles, triangles, circles, boxes, cylinders and spheres.

The Idea Behind It

A triangle's area is half the base times the height. Where does the $\frac{1}{2}$ come from? *but why?*

The rectangle is easy: a b by h rectangle holds $b \times h$ unit squares, so its area is $b \times h$. A triangle gets a stubborn-looking $\frac{1}{2}$ in front. That factor is not a fudge, it is forced, and the rectangle is exactly where it comes from:

$$\boxed{\text{area of a rectangle} = \text{base} \times \text{height}}$$

Take any triangle on base b with height h. Make a second copy of it, turn that copy upside down (a half-turn), and slot it against the first. The two together fill a complete **parallelogram** standing on the same base b with the same height h:

$$\text{two triangles} = \text{a parallelogram of area } b \times h$$

A parallelogram has the same area as the rectangle with that base and height (slide the slanted end straight across, no area is gained or lost). So two triangles cover $b \times h$, which means **one** triangle covers half of it:

$$\text{area}_{\triangle} = \frac{b \times h}{2} = \tfrac{1}{2}\, b\, h$$

$$\therefore \text{ a triangle is } \text{half its bounding rectangle}$$

That is the whole story behind the $\frac{1}{2}$: every triangle pairs up with a rotated twin to box in a rectangle, and it owns exactly half.

the height must be the perpendicular drop to the base, not a slanted side

Every physical thing has a size we can measure. How much fencing surrounds a garden? How much paint covers a wall? How much water fills a tank? These three questions are really three different *kinds* of measurement, and almost all of geometry's practical power comes from knowing which one you need.

There are three rungs on the ladder, and they go up one **dimension** at a time:

PERIMETER, AREA, VOLUME

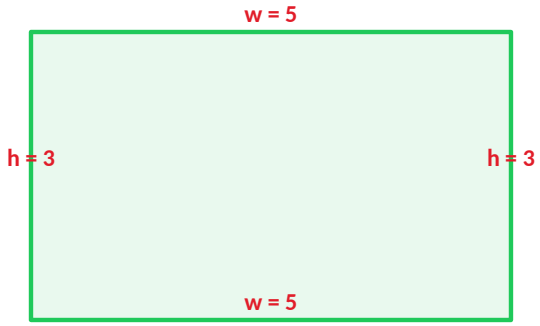
Perimeter is the distance *around* a flat shape – add up the lengths of its edges (measured in metres, m). **Area** is the amount of flat space *inside* it (measured in square metres, m²). **Volume** is the amount of space a solid *fills* (measured in cubic metres, m³).

perimeter area volume surface area

Flat shapes: perimeter and area

For a **rectangle** with width w and height h, the perimeter is twice the width plus twice the height, and the area is width times height:

$$P = 2w + 2h, \qquad A = w \times h$$

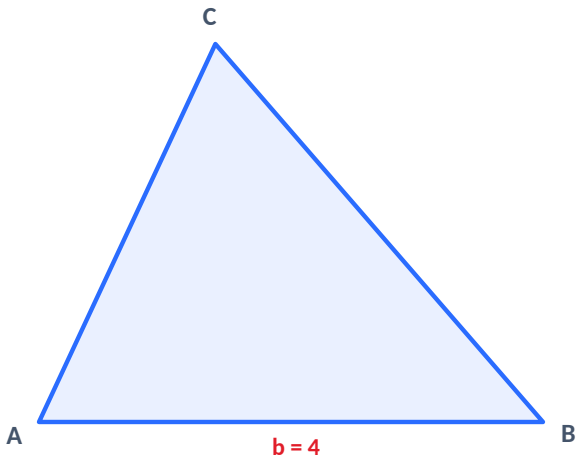


A 5 by 3 rectangle. Perimeter = 2(5)+2(3) = 16. Area = 5×3 = 15 squares.

Why does area count *squares*? Lay a grid of 1×1 squares over the rectangle: a 5-by-3 rectangle is exactly 5 columns and 3 rows of them, so $5 \times 3 = 15$ unit squares fit inside. That is what "15 square metres" means – fifteen 1 m × 1 m tiles.

For a **triangle**, the area is half the base times the height (the height is the straight-up distance from the base to the opposite tip, meeting the base at a right angle):

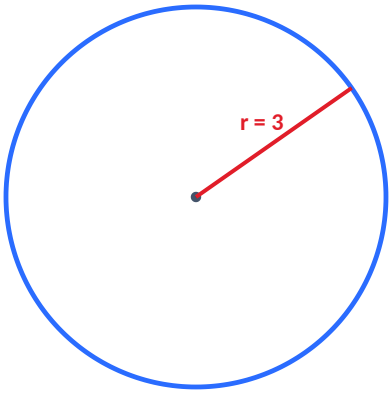
$$A_{\triangle} = \tfrac{1}{2}\, b \times h$$



A triangle on base b = 4 with height h = 3 (the dashed-style vertical drop). Area = ½ × 4 × 3 = 6.

A triangle is exactly **half** of a rectangle that boxes it in, which is where the $\frac{1}{2}$ comes from. For a **circle** of radius r (the distance from the centre to the edge), we use the special number $\pi \approx 3.14159$:

$$C = 2\pi r \quad (\text{the perimeter, called the } \mathbf{circumference}), \qquad A = \pi r^2$$

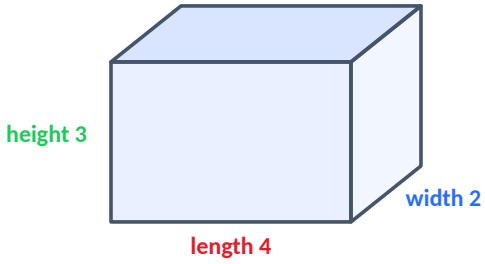


A circle of radius 3. Circumference = 2π(3) ≈ 18.85. Area = π(3²) = 9π ≈ 28.27.

Solid shapes: surface area and volume

Step up to three dimensions. A **rectangular box** (a "cuboid") with length ℓ, width w and height h has a volume of length × width × height – literally how many 1×1×1 cubes pack inside:

$$V_{\text{box}} = \ell \times w \times h$$



A 4 × 2 × 3 box. Volume = 4×2×3 = 24 cubic units. It has 6 rectangular faces.

The box's **surface area** is the total of its six faces. Opposite faces match, so we add the three distinct faces and double:

$$S_{\text{box}} = 2(\ell w + \ell h + w h)$$

A **cylinder** (a tube, like a can) of radius r and height h stacks circles of area πr^2 to height h, so its volume is the circle area times the height:

$$V_{\text{cylinder}} = \pi r^2 h$$

A **sphere** (a ball) of radius r has these two formulas – worth memorising, as they appear constantly in science and graphics:

$$V_{\text{sphere}} = \tfrac{4}{3}\pi r^3, \qquad S_{\text{sphere}} = 4\pi r^2$$

Note: Watch the units. A length is in m, an area in m² (a length squared), a volume in m³ (a length cubed). If a formula multiplies three lengths together, the answer must be a volume – a quick way to check you used the right one.

Worked examples

Example 1

A rectangular garden is 8 m wide and 5 m high (deep). Find its perimeter and area.

SOLUTION

1. Perimeter is the distance all the way round: $P = 2w + 2h = 2(8) + 2(5)$. round the edge
2. $= 16 + 10 = 26$ m of fencing.
3. Area is the space inside: $A = w \times h = 8 \times 5 = 40$.

Perimeter $= 26$ m; area $= 40$ m².

Example 2

Find the area of a triangle with base 10 cm and height 6 cm.

SOLUTION

1. Use $A = \tfrac{1}{2}bh$ with $b = 10$ and $h = 6$. ÷2 for area
2. $A = \tfrac{1}{2} \times 10 \times 6 = \tfrac{1}{2} \times 60$.
3. $= 30$.

Area $= 30$ cm².

Example 3

A circular pizza has radius $r = 15$ cm. Find its circumference and area (use $\pi \approx 3.14$).

SOLUTION

1. Circumference (distance round the crust): $C = 2\pi r = 2 \times 3.14 \times 15$.
2. $= 6.28 \times 15 = 94.2$ cm.
3. Area: $A = \pi r^2 = 3.14 \times 15^2 = 3.14 \times 225$. square r
4. $= 706.5$.

Circumference ≈ 94.2 cm; area ≈ 706.5 cm².

Example 4

A box is $\ell = 4$, $w = 2$, $h = 3$ (in metres). Find its volume and surface area.

SOLUTION

1. Volume: $V = \ell w h = 4 \times 2 \times 3 = 24$ m³. *l × w × h*
2. Surface area uses the three distinct faces: $\ell w = 8$, $\ell h = 12$, $wh = 6$.
3. Add them: $8 + 12 + 6 = 26$, then double for the matching opposite faces. *×2 opposite faces*
4. $S = 2 \times 26 = 52$.

Volume $= 24$ m³; surface area $= 52$ m².

Example 5

A cylindrical water tank has radius $r = 2$ m and height $h = 5$ m. Find its volume (use $\pi \approx 3.14$).

SOLUTION

1. Volume of a cylinder: $V = \pi r^2 h$. *circle × height*
2. First the base circle: $r^2 = 2^2 = 4$, so the base area is $\pi \times 4 = 3.14 \times 4 = 12.56$ m².
3. Multiply by the height: $V = 12.56 \times 5$.
4. $= 62.8$.

Volume ≈ 62.8 m³ (about $62{,}800$ litres of water).

Example 6

A ball (sphere) has radius $r = 3$ cm. Find its volume (use $\pi \approx 3.14$).

SOLUTION

1. Volume of a sphere: $V = \frac{4}{3}\pi r^3$. *cube the r*
2. Cube the radius: $r^3 = 3^3 = 27$.
3. Now $\frac{4}{3} \times \pi \times 27 = 4 \times \pi \times 9 = 36\pi$.
4. $\approx 36 \times 3.14 = 113.04$.

Volume ≈ 113.0 cm³.

Example 7

A swimming pool is a box 25 m long, 10 m wide and 2 m deep. How many litres of water fills it? (1 m³ = 1000 litres.)

SOLUTION

1. First the volume in cubic metres: $V = 25 \times 10 \times 2 = 500$ m³. *l × w × h*
2. Convert to litres: each m³ is 1000 litres, so 500×1000. *×1000 → litres*
3. $= 500{,}000$.

$500{,}000$ **litres (half a million litres).**

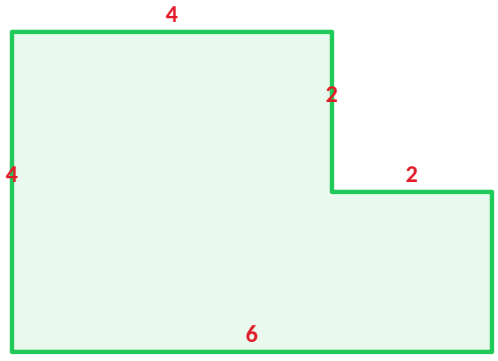
Example 8

An L-shaped room is made by removing a 2 m by 2 m corner from a 6 m by 4 m rectangle. Find its floor area.

SOLUTION

1. Two ways work; the quickest is "whole minus the bite". Start with the full rectangle: $6 \times 4 = 24$ m². *full rectangle*
2. The missing corner is a small square: $2 \times 2 = 4$ m². *the bite*
3. Subtract the bite from the whole: $24 - 4 = 20$ m².

The floor area is 20 m². (Split-and-add gives the same: a 6×2 strip plus a 4×2 strip $= 12 + 8 = 20$.)



The L-shaped floor: a 6×4 rectangle with a 2×2 corner removed. Area = 24 − 4 = 20 m² (or split into a 6×2 and a 4×2 strip).

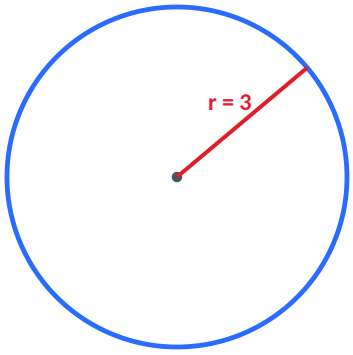
Example 9

A closed cylindrical can has radius $r = 3$ cm and height $h = 10$ cm. Find its total **surface area** (use $\pi \approx 3.14$).

SOLUTION

1. A can has three pieces of skin: a top circle, a bottom circle, and the curved side. The two circles each have area $\pi r^2 = 3.14 \times 3^2 = 3.14 \times 9 = 28.26$ cm². *two end caps*
2. Unroll the curved side and it becomes a rectangle: its width is the circumference $2\pi r$ and its height is h, so its area is $2\pi rh = 2 \times 3.14 \times 3 \times 10 = 188.4$ cm². *side unrolls to a rectangle*
3. Add the two caps and the side: $28.26 + 28.26 + 188.4 = 244.92$ cm².

The total surface area is ≈ 244.92 cm² (two circles + one unrolled rectangle).



One circular end-cap of the can (area πr²). The can has two of these, plus a curved side that unrolls into a rectangle of width 2πr and height h.

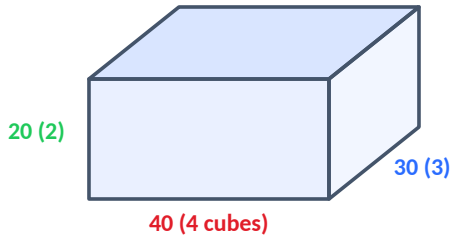
Example 10

A storage crate measures 40 cm by 30 cm by 20 cm. How many 10 cm cubes fit inside it?

SOLUTION

1. Each 10 cm cube has volume $10 \times 10 \times 10 = 1000$ cm³. *one cube*
2. The crate volume is $40 \times 30 \times 20 = 24{,}000$ cm³. *l × w × h*
3. How many cubes fit is "crate volume ÷ cube volume": $24{,}000 \div 1000 = 24$. *÷ to count cubes*

Exactly 24 cubes fit – 4 along, 3 across and 2 high, since $4 \times 3 \times 2 = 24$.



The crate is 4 cubes long, 3 cubes deep and 2 cubes high, so 4×3×2 = 24 of the 10 cm cubes pack inside.

Mental-Math Shortcut

Area of a circle, the quick way. π is just a hair over 3, so for a rough head-figure use $A \approx 3r^2$ and round up a touch. A circle of radius 4: $3 \times 16 = 48$, nudge up to about 50 (the exact value is 50.3).

Doubling a dimension multiplies area by 4, volume by 8. Double a square's side and you fit *four* of the originals inside (2^2); double a cube's side and *eight* fit (2^3). So a pizza of twice the radius is four times the food – useful when comparing prices.

Why is the area of a rectangle *exactly* width × height, and not something you have to measure square by square? The argument is a clean piece of counting.

Take a rectangle that is a whole number of units wide and tall, say $w = 5$ and $h = 3$. Lay down the unit squares in rows. **Each row** sits along the width, so it holds exactly $w = 5$ squares. There are exactly $h = 3$ such rows stacked up. Counting every square is therefore "the same number, added h times":

$$\underbrace{w + w + w}_{h \text{ rows}} = h \times w.$$

That is just what multiplication *means* – repeated addition of equal groups. So the area is forced to be $w \times h$; there is no other possibility once you accept the grid.

The same logical move climbs a dimension: a box is a stack of h identical flat layers, each layer a rectangle of area $\ell \times w$, giving volume $\ell \times w \times h$. *One rule, applied one dimension at a time.* (For shapes with curved or fractional sides the count is taken to a limit – that is exactly the integral idea of Chapter 58.)

⚠ Common Mistakes

- Mixing up area and perimeter. Perimeter is a **length** (add the edges, answer in m); area is the **space inside** (answer in m²). A bigger area does not always mean a bigger perimeter.
- Forgetting the $\frac{1}{2}$ in a triangle. The triangle is half its bounding rectangle: $A = \frac{1}{2}bh$, not bh.
- Squaring the diameter instead of the radius in a circle. Area is πr^2 with r the **radius** (half the diameter) – using the diameter makes the answer four times too big.
- Writing the answer in the wrong units: a volume must be in cubic units (m³), an area in square units (m²). If your "volume" is in m², you have multiplied only two lengths.
- Cubing wrongly in the sphere formula: r^3 means $r \times r \times r$. For $r = 3$ that is 27, not 9.

Quiz

49.1.Q1. A rectangle is 7 m by 4 m. What is its **area**?

A. 22 m²

B. 28 m²

C. 11 m²

D. 28 m

49.1.Q2. A rectangle is 7 m by 4 m. What is its **perimeter**?

49.1.Q3. Find the area of a triangle with base 8 and height 5.

A. 40

B. 20

C. 13

D. 26

49.1.Q4. A circle has radius $r = 10$. Write its area in terms of π.

49.1.Q5. A box is $3 \times 3 \times 3$. What is its volume?

A. 9

B. 27

C. 18

D. 54

49.1.Q6. A cylinder has base area 12 m² and height 4 m. What is its volume?

49.1.Q7. True or False: a sphere of radius r has volume $\frac{4}{3}\pi r^3$.

A. True

B. False

49.1.Q8. If you double every side of a cube, the volume becomes how many times as large?

A. 2

B. 4

C. 6

D. 8

🐍 From Concept to Code (Python)

Each formula becomes a one-line function. Using `math.pi` for π, we can build a tiny library that measures any of these shapes – and a computer never forgets the $\frac{1}{2}$ or mixes up the units.

```
import math

def rect_area(w, h):
    return w * h

def circle_area(r):
    return math.pi * r ** 2

def box_volume(l, w, h):
    return l * w * h

print(rect_area(5, 3))
print(round(circle_area(3), 2))
print(box_volume(4, 2, 3))
```

```
output
15
28.27
24
```

```
import math

def sphere_volume(r):
    return (4 / 3) * math.pi * r ** 3

def cylinder_volume(r, h):
    return math.pi * r ** 2 * h

print(round(sphere_volume(3), 2))     # a ball, radius 3
print(round(cylinder_volume(2, 5), 2))  # a tank, r=2 h=5
```

```
output
113.1
62.83
```

Try it:

1. Write `triangle_area(b, h)` returning $\frac{1}{2}bh$, then print the area of a triangle with base 10 and height 6.

2. Use `box_volume` to find the litres in a $25 \times 10 \times 2$ m pool (multiply the m³ result by 1000). Do you get 500000?

Answer Key – Chapter 49 – Distance, Area & Volume

49.1.Q1. 28 m² – area is width × height $= 7 \times 4 = 28$, in square metres.

49.1.Q2. 22 m – $2(7) + 2(4) = 14 + 8 = 22$ m of edge.

49.1.Q3. 20 – $A = \frac{1}{2} \times 8 \times 5 = 20$.

49.1.Q4. 100π – $A = \pi r^2 = \pi(10^2) = 100\pi \approx 314$.

49.1.Q5. 27 – $3 \times 3 \times 3 = 27$ cubic units.

49.1.Q6. 48 m³ – volume is base area × height $= 12 \times 4 = 48$.

49.1.Q7. True – that is the sphere volume formula; its surface area is $4\pi r^2$.

49.1.Q8. 8 – volume scales by $2^3 = 8$ when each length doubles.

Chapter 50 · Transformations & Geometric Vectors

How to move a shape on the coordinate plane without changing its rule – sliding it (translation), flipping it (reflection), turning it (rotation) and resizing it (scaling) – by doing simple arithmetic on the coordinates of its corners.

The Idea Behind It

Sliding by $\vec{u}$ then by $\vec{v}$ is the same single slide $\vec{u}+\vec{v}$, and the order makes no difference. *but why?*

A translation just *adds a fixed arrow* to every point: slide by $\vec{u}$ and the point p goes to $p+\vec{u}$. Do a second slide $\vec{v}$ and it lands at $(p+\vec{u})+\vec{v}$. Two slides in a row are therefore nothing more than two additions in a row, and how additions stack is governed by one rule you already trust for numbers, the **associative + commutative law of addition**:

$$\boxed{(p+\vec{u})+\vec{v} = p+(\vec{u}+\vec{v}), \qquad \vec{u}+\vec{v} = \vec{v}+\vec{u}}$$

Regroup the brackets with the **associative** law: the two arrows join into one combined arrow $\vec{u}+\vec{v}$ added to p just once.

$$(p+\vec{u})+\vec{v} = p+\ (\vec{u}+\vec{v})\ = p+(\vec{u}+\vec{v})$$

Now swap the order of the slides. Because each coordinate adds independently and number addition **commutes** ($a+b=b+a$ in every component), the combined arrow is identical either way:

$$\vec{v}+\vec{u} = \vec{u}+\vec{v} \quad \Rightarrow \quad (p+\vec{v})+\vec{u} = p+(\vec{u}+\vec{v})$$

$$\therefore\ \text{slide } \vec{u} \text{ then } \vec{v}\ =\ \text{slide } \vec{v} \text{ then } \vec{u}\ =\ \text{slide } (\vec{u}+\vec{v})$$

So a chain of translations always collapses to a single arrow, the sum of the parts, and reshuffling the steps cannot change where you end up.

true only because translations are pure addition; rotations are not, so their order can matter

A **transformation** moves every point of a shape to a new place by a fixed rule. This is exactly how a computer animates a character walking, a photo editor flips an image, or a game spins a spaceship: it does not redraw the shape, it just *moves each corner* with a little arithmetic and reconnects them. We will meet the four most important moves.

TRANSFORMATION

A **transformation** is a rule that sends each point (x, y) to a new point (x', y'). The original is the **pre-image**; the result is the **image**. We write the corner A and its moved copy A' (read "A prime").

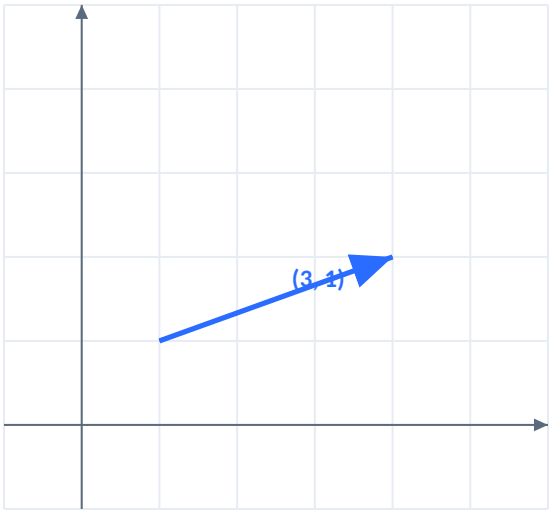
transformation image translation vector

1. Translation – a slide (and it IS a vector)

A **translation** slides the whole shape: every point moves the *same* amount across and the *same* amount up. If you move a to the right and b up, then

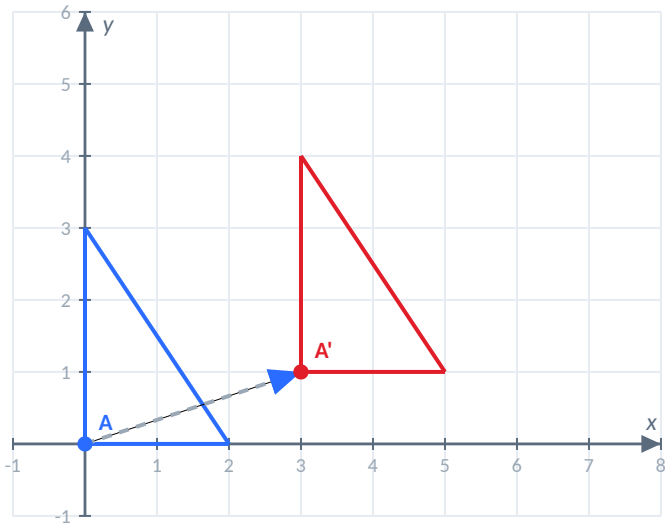
$$(x, y) \longrightarrow (x', y') = (x + a,\ y + b)$$

That pair (a, b) – "go a across, b up" – is precisely a **vector**: an arrow describing a displacement. So a translation *is* a vector applied to every point. Here is the vector $(3, 1)$ that performs the slide:



The translation vector (3, 1): every point moves 3 right and 1 up. Tail at a point, head where it lands.

Watch a triangle slide by this vector. Each corner just has $(3, 1)$ added to it; the shape keeps its exact size and orientation:

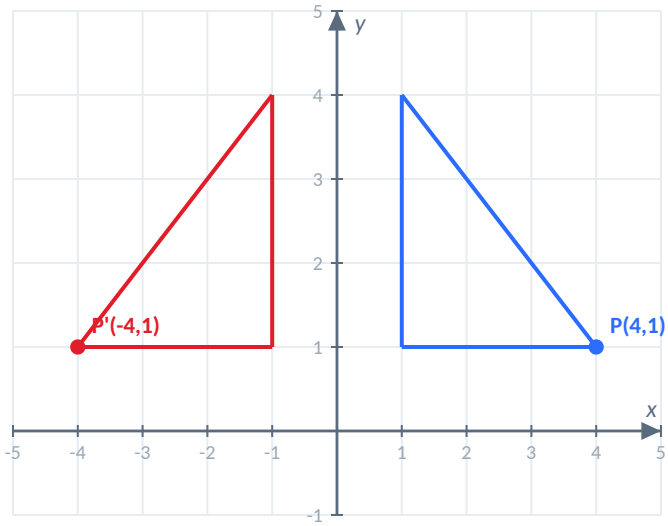


Blue pre-image slid by (3, 1) to the red image. Same size, same tilt – just shifted.

2. Reflection – a mirror flip

A **reflection** flips the shape across a mirror line. Across the two axes the rule is simply a sign change:

- Across the x**-axis** (horizontal mirror): $(x, y) \to (x, -y)$ – the height flips sign.
- Across the y**-axis** (vertical mirror): $(x, y) \to (-x, y)$ – the across-value flips sign.



Reflection across the y-axis: each x flips sign, y stays. The image is the mirror twin.

3. Rotation – a turn about the origin

A **rotation** turns the shape about a fixed centre. About the origin, the quarter-turns have tidy rules (anticlockwise is the positive direction):

- 90° anticlockwise: $(x, y) \to (-y, x)$.
- 180° (half turn): $(x, y) \to (-x, -y)$.
- 270° anticlockwise: $(x, y) \to (y, -x)$.

4. Scaling – resize from the origin

A **scaling** (or dilation) by a factor k stretches every coordinate away from the origin by k:

$$(x, y) \longrightarrow (kx,\ ky)$$

If $k > 1$ the shape grows; if $0 < k < 1$ it shrinks; the shape keeps its proportions (it stays *similar*).

> **Note:** Translation, reflection and rotation keep the shape exactly the same size and form – these are called **rigid** motions (think of sliding a rigid tile around). Scaling is the odd one out: it changes the size.

Worked examples

Example 1

Translate the point $A(2, 5)$ by the vector $(3, -2)$.

SOLUTION

1. A translation **adds** the vector to the point: $(x + a,\ y + b)$. *slide each part*
2. Add across: $2 + 3 = 5$. Add up: $5 + (-2) = 3$.

$A' = (5, 3)$.

Example 2

A triangle has corners $A(0, 0)$, $B(2, 0)$, $C(0, 3)$. Slide it by $(3, 1)$.

SOLUTION

1. Apply $(x + 3,\ y + 1)$ to **each** corner separately. *+3 across, +1 up*
2. $A(0, 0) \to (3, 1)$.
3. $B(2, 0) \to (5, 1)$.
4. $C(0, 3) \to (3, 4)$.

$A'(3, 1)$, $B'(5, 1)$, $C'(3, 4)$ **– exactly the red triangle drawn above.**

Example 3

Reflect the point $P(4, 1)$ across the y-axis.

SOLUTION

1. Reflecting across the y-axis flips the sign of x and keeps y: $(x, y) \to (-x, y)$. *x → −x*
2. So $x : 4 \to -4$, and $y : 1 \to 1$ (unchanged).

$P' = (-4, 1)$.

Example 4

Reflect the point $Q(3, 2)$ across the x-axis.

SOLUTION

1. Reflecting across the x-axis flips the sign of y: $(x, y) \to (x, -y)$. *y → −y*
2. So x stays 3, and $y : 2 \to -2$.

$Q' = (3, -2)$.

Example 5

Rotate the point $R(4, 1)$ by 90° anticlockwise about the origin.

SOLUTION

1. The 90° rule is $(x, y) \to (-y, x)$ – swap, then minus the first. *swap + flip sign*
2. New x is $-y = -1$. New y is $x = 4$.

$R' = (-1, 4)$.

Example 6

Rotate the point $S(2, 5)$ by 180° about the origin.

SOLUTION

1. A half turn is $(x, y) \to (-x, -y)$. *flip both signs*
2. Negate both: $x: 2 \to -2$, $y: 5 \to -5$.

$S' = (-2, -5)$.

Example 7

Scale the triangle $A(1, 1)$, $B(2, 1)$, $C(1, 3)$ from the origin by factor $k = 2$.

SOLUTION

1. Scaling by 2 **doubles** both coordinates of every point: $(x, y) \to (2x, 2y)$. *×2, do not add*
2. $A(1, 1) \to (2, 2)$.
3. $B(2, 1) \to (4, 2)$.
4. $C(1, 3) \to (2, 6)$.

$A'(2, 2)$, $B'(4, 2)$, $C'(2, 6)$ **– the same shape, twice as large.**

Example 8

A translation sends $A(2, 3)$ to $A'(7, 1)$. Find the translation vector, then use it to move $B(0, 4)$.

SOLUTION

1. A translation adds a fixed vector (a, b), so (a, b) is image **minus** pre-image: subtract the start from the end.
   end – start
2. Across: $a = 7 - 2 = 5$. Up: $b = 1 - 3 = -2$, so the vector is $(5, -2)$.
3. Apply that same vector to B: across $0 + 5 = 5$, up $4 + (-2) = 2$.

The vector is $(5, -2)$, and it sends B to $B' = (5, 2)$.

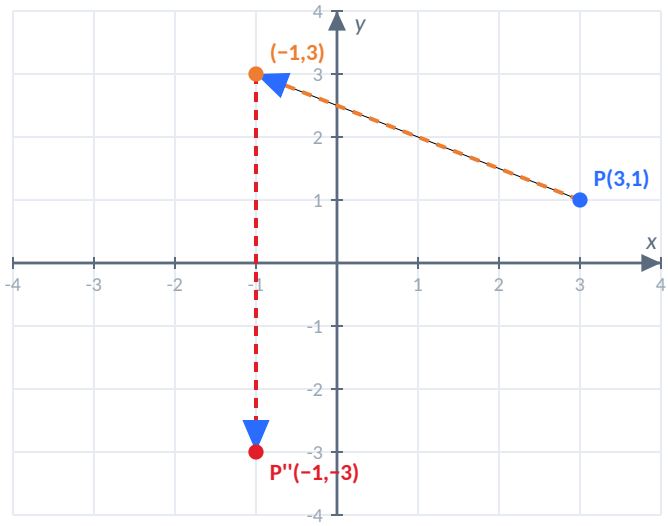
Example 9

Rotate the point $P(3, 1)$ by 90° anticlockwise about the origin, then **reflect** the result across the x-axis. Where does P land?

SOLUTION

1. Do the moves in order, left to right – the second acts on the result of the first. *order matters*
2. Rotate 90°: the rule is $(x, y) \to (-y, x)$, so $P(\ 3, 1\ ) \to (\ -1, 3\ )$. *swap + flip sign*
3. Reflect that point across the x-axis: $(x, y) \to (x, -y)$, so $(\ -1, 3\ ) \to (\ -1, -3\ )$. *y → -y*

P **ends at** $(\ -1, -3\ )$**. The diagram below traces the two hops.**



P(3,1) rotates 90° anticlockwise to (−1,3) (orange hop), then reflects across the x-axis to (−1,−3) (red hop).

Example 10

A triangle has corners $A(1, 1)$, $B(3, 1)$, $C(1, 2)$. Reflect it across the y-axis, then translate the result by $(0, 3)$. Give the final corners.

SOLUTION

1. Reflect each corner across the y-axis first: $(x, y) \to (-x, y)$ (negate the across-value). *x → -x*
2. $A(\ 1, 1\ ) \to (-1, 1)$, $B(\ 3, 1\ ) \to (-3, 1)$, $C(\ 1, 2\ ) \to (-1, 2)$.
3. Now translate each by $(0, 3)$ – add 3 to every up-value: $(x, y) \to (x, y + 3)$. *slide up 3*
4. $(-1,\ 1\ ) \to (-1,\ 4\ )$, $(-3,\ 1\ ) \to (-3,\ 4\ )$, $(-1,\ 2\ ) \to (-1,\ 5\ )$.

$A''(\ -1, 4\ )$, $B''(\ -3, 4\ )$, $C''(\ -1, 5\ )$ **– mirrored, then lifted three units.**

⚡ Mental-Math Shortcut

You do not need to memorise the rotation rules as formulas – read them off your hand. A 90° anticlockwise turn sends the "right" direction to "up": so the **new across-value is the old up-value with a minus**, and the **new up-value is the old across-value**, giving $(x, y) \to (-y, x)$. A half turn (180°) just points everything the opposite way, so **flip both signs**: $(x, y) \to (-x, -y)$. And reflections are a one-word check: "across the x-axis" flips the *other* coordinate, y; "across the y-axis" flips x.

Doing two transformations in a row is itself a single transformation – and the order can matter. This is a small but important piece of logical reasoning, because it is the same structure as **composing functions**.

Take the point $P(4, 1)$. **Path A:** reflect across the y-axis, *then* across the x-axis.

$$(4, 1) \xrightarrow{y\text{-axis}} (-4, 1) \xrightarrow{x\text{-axis}} (-4, -1).$$

Path B: do the two reflections in the *opposite* order.

$$(4, 1) \xrightarrow{x\text{-axis}} (4, -1) \xrightarrow{y\text{-axis}} (-4, -1).$$

Here both paths land on $(-4, -1)$ – and that point is exactly $(-x, -y)$, a 180° rotation. So *two perpendicular reflections equal one half-turn*, an "if both, then" conclusion you can prove for any point.

But order is **not** always harmless. Reflect $P(4, 1)$ across the y-axis then rotate 90°, versus rotate first then reflect, and you reach different points. The lesson: a sequence of moves is read **left to right like the steps of a proof** – each operates on the result of the one before, and swapping steps can change the conclusion.

⚠ Common Mistakes

- Translating only one corner of a shape. A translation moves **every** point by the same vector – add (a, b) to all the corners, then reconnect.
- Flipping the wrong coordinate in a reflection. Across the **x-axis** you negate y (the up-value); across the **y-axis** you negate x. It is the coordinate *perpendicular* to the mirror that changes.
- Confusing the 90° rotation rule. Anticlockwise is $(x, y) \to (-y, x)$ – the coordinates swap **and** the new first one gets a minus. Forgetting the minus rotates the wrong way.
- Adding instead of multiplying when scaling. Scaling by k **multiplies** both coordinates (kx, ky); it does not add k.
- Assuming order never matters. A reflection followed by a rotation can land somewhere different from the rotation followed by the reflection – do them in the stated order.

Quiz

50.1.Q1. Translate $(2, 7)$ by the vector $(4, -3)$.

A. $(6, 4)$

B. $(8, -21)$

C. $(-2, 10)$

D. $(6, 10)$

50.1.Q2. What single object describes a translation?

A. a vector

B. a circle

C. an angle

D. a matrix

50.1.Q3. Reflect $(5, 2)$ across the x-axis.

50.1.Q4. Reflect $(5, 2)$ across the y-axis.

50.1.Q5. Rotate $(3, 0)$ by 90° anticlockwise about the origin.

A. $(0, 3)$

B. $(0, -3)$

C. $(-3, 0)$

D. $(3, 0)$

50.1.Q6. Rotate $(2, 5)$ by 180° about the origin.

50.1.Q7. Scale $(3, 4)$ from the origin by factor $k = 2$.

50.1.Q8. True or False: a reflection changes the size of a shape.

A. True

B. False

🐍 From Concept to Code (Python)

A point is just a tuple `(x, y)`, and each transformation is a short function that returns the new tuple. This is the literal arithmetic a graphics program runs on every corner of every shape, many times a second.

```
def translate(p, a, b):
    x, y = p
    return (x + a, y + b)      # slide by vector (a, b)

def reflect_y(p):
    x, y = p
    return (-x, y)             # mirror across the y-axis

def rotate90(p):
    x, y = p
    return (-y, x)             # 90 deg anticlockwise

print(translate((2, 5), 3, -2))
print(reflect_y((4, 1)))
print(rotate90((4, 1)))
```

```
output
(5, 3)
(-4, 1)
(-1, 4)
```

```
# Transform a whole shape: apply the rule to every corner
tri = [(0, 0), (2, 0), (0, 3)]
image = [translate(corner, 3, 1) for corner in tri]
print(image)
```

```
output
[(3, 1), (5, 1), (3, 4)]
```

Try it:

1. Write `scale(p, k)` that returns `(k*x, k*y)`, then scale the corners of `tri` by 2.
2. Write `reflect_x(p)` (flip the y-value) and apply it to every corner of `tri` with a list comprehension.

Answer Key – Chapter 50 – Transformations & Geometric Vectors

50.1.Q1. $(6, 4)$ – add the vector: $2 + 4 = 6$ and $7 + (-3) = 4$.

50.1.Q2. A vector – the pair (a, b) "go a across, b up" is exactly a translation.

50.1.Q3. $(5, -2)$ – across the x-axis the y-value flips sign.

50.1.Q4. $(-5, 2)$ – across the y-axis the x-value flips sign.

50.1.Q5. $(0, 3)$ – the rule $(x, y) \to (-y, x)$ gives $(-0, 3) = (0, 3)$.

50.1.Q6. $(-2, -5)$ – a half turn negates both coordinates.

50.1.Q7. $(6, 8)$ – multiply both coordinates by 2.

50.1.Q8. False – reflection (like translation and rotation) is rigid; only scaling changes size.

Chapter 51 · Projections

How three-dimensional objects are drawn on a flat page – orthographic and isometric views – and how to "project" a point by dropping one of its coordinates, the trick that turns 3-D world positions into 2-D screen positions.

The Idea Behind It

How far does $\vec{a}$ reach *along* $\vec{b}$? The answer is the scalar projection $\frac{\vec{a}\cdot\vec{b}}{|\vec{b}|}$. but why?

Drop $\vec{a}$ straight down onto the line of $\vec{b}$. The shadow it casts is the **adjacent** side of a right triangle whose hypotenuse is $\vec{a}$ and whose angle to $\vec{b}$ is θ. Plain trigonometry ($\cos = \frac{\text{adjacent}}{\text{hypotenuse}}$) gives the length of that shadow at once:

$$\text{shadow length} = |\vec{a}|\cos\theta$$

That is already the projection, but θ is awkward to measure. So we lean on the rule that *defines* the **dot product**, which packages exactly this $|\vec{a}|\cos\theta$ together with $|\vec{b}|$:

$$\boxed{\vec{a}\cdot\vec{b} = |\vec{a}|\,|\vec{b}|\cos\theta}$$

The shadow we want is $|\vec{a}|\cos\theta$, and the box hands us $|\vec{a}|\,|\vec{b}|\cos\theta$, which is just that shadow scaled up by $|\vec{b}|$. Divide the dot product by $|\vec{b}|$ to cancel the extra factor:

$$\frac{\vec{a}\cdot\vec{b}}{|\vec{b}|} = \frac{|\vec{a}|\;|\vec{b}|\;\cos\theta}{|\vec{b}|} = |\vec{a}|\cos\theta = \text{shadow length}$$

$$\therefore\ \text{scalar projection of } \vec{a} \text{ onto } \vec{b} = \frac{\vec{a}\cdot\vec{b}}{|\vec{b}|}$$

So the dot product is secretly a projection in disguise: divide it by the length of $\vec{b}$ and you read off how far $\vec{a}$ stretches in $\vec{b}$'s direction, with no angle ever measured.

if $\vec{b}$ is already a unit vector, $|\vec{b}| = 1$ and the projection is simply $\vec{a}\cdot\vec{b}$

The world has three dimensions, but a page, a photograph and a computer screen are flat – they have only two. So every drawing of a solid object is a kind of clever lie: it shows a *flattened* version. The rules for flattening sensibly are called **projection**, and the same idea lets a game turn a character's 3-D position into a dot on your 2-D screen.

PROJECTION

A **projection** represents a higher-dimensional object in fewer dimensions by *discarding* or *combining* some of the information – most simply, by **dropping one coordinate**. Projecting from 3-D to 2-D throws away one of the three numbers (x, y, z).

projection orthographic view isometric view

Orthographic views – straight-on snapshots

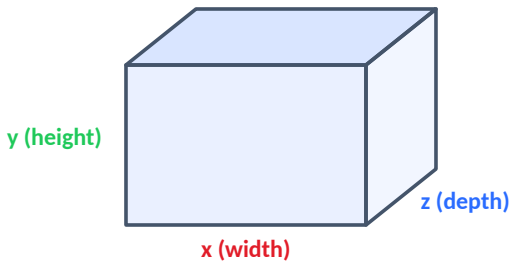
The simplest way to record a solid is to look at it *straight on* from a few directions and draw what you see. An **orthographic** view shows one face flat, with no shrinking-into-the-distance. Engineers draw three: the **front**, the **top** (from above) and the **side**. Each is a true-shape rectangle; together they pin down the object exactly.

Mathematically, each orthographic view is a projection that **drops one coordinate**:

- The **front** view looks along the depth axis, so it keeps (x, y) and drops z.
- The **top** view looks straight down, keeping (x, z) and dropping y.
- The **side** view looks along x, keeping (z, y) and dropping x.

Isometric view – a 3-D look on flat paper

Orthographic views are exact but never *look* solid. An **isometric** drawing fakes depth: the three directions (width, height, depth) are drawn at equal 120° angles, so a box looks three-dimensional while edges of equal length stay equal on the page. This is the classic look of building blocks and many video games.

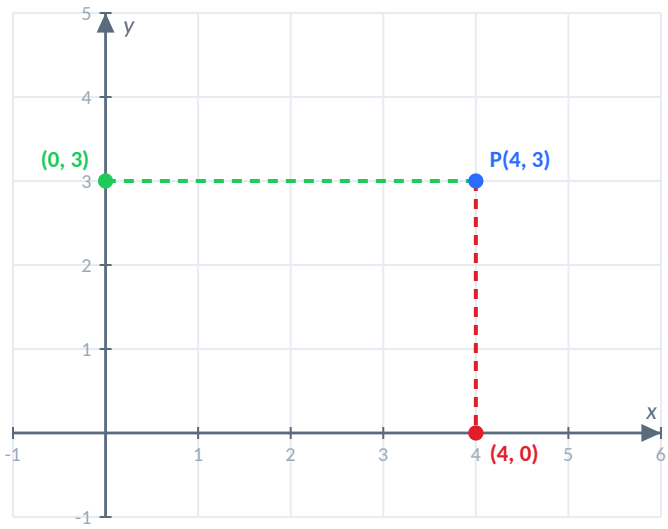


An isometric view of a box. The depth axis z is drawn slanting up-right, giving a solid look without a vanishing point.

(A **perspective** view goes one step further and makes distant edges *shrink*, the way railway tracks seem to meet on the horizon. That extra realism is why photos use it – but isometric keeps the maths simple, which is why so many games choose it.)

Projecting a single point onto an axis

The most basic projection of all takes a point and asks: *what is its shadow on one axis?* To project $P(x, y)$ onto the x**-axis**, you simply read off its x-coordinate and set the height to 0 – you drop a straight line down to the axis. The foot of that line, $(x, 0)$, is the projection.



Dropping P(4,3) onto the x-axis gives (4,0); onto the y-axis gives (0,3). Each projection keeps one coordinate and zeroes the other.

So projecting onto an axis is just **keeping one coordinate**. Projecting a 3-D point onto the 2-D screen is the same move one size up: keep two coordinates, drop the third. The simplest screen projection drops the depth z:

$$(x,\ y,\ z) \longrightarrow (x,\ y)$$

Note: Dropping a coordinate loses information you cannot recover. Many different 3-D points share the same flat shadow – the point $(4, 3, 1)$ and $(4, 3, 9)$ both project to $(4, 3)$. That is exactly why a single photo cannot tell you how far away something is.

Worked examples

Example 1

Project the 3-D point $P(4, 3, \ 7\ )$ onto the screen by dropping the depth z.

SOLUTION

1. Screen projection keeps the first two coordinates and discards z: $(x, y, z) \to (x, y)$. *drop z*
2. Keep $x = 4$ and $y = 3$; throw away $z = 7$.

The screen position is $(\ 4, 3\ )$.

Example 2

Project the point $Q(\ 4\ ,\ 3\ )$ onto the x-axis.

SOLUTION

1. Projecting onto the x-axis keeps the x-coordinate and sets y to 0. *keep x, zero y*
2. Drop a vertical line from $(4, 3)$ straight down; it meets the axis at height 0.

The projection is $(\ 4\ ,\ 0\ )$.

Example 3

Project the same point $Q(\ 4\ ,\ 3\ )$ onto the y-axis.

SOLUTION

1. Projecting onto the y-axis keeps the y-coordinate and sets x to 0. *keep y, zero x*
2. Drop a horizontal line from $(4, 3)$ across to the axis; it meets it at across-value 0.

The projection is $(\ 0\ ,\ 3\ )$**.**

Example 4

A cube corner sits at $(2, 5, 3)$. Give its front, top and side orthographic projections.

SOLUTION

1. Front view drops the depth z: keep $(x, y) = (\ 2, 5\ )$. *drop z*
2. Top view (looking down) drops the height y: keep $(x, z) = (\ 2, 3\ )$. *drop y*
3. Side view drops the width x: keep $(z, y) = (\ 3, 5\ )$. *drop x*

Front $(\ 2, 5\ )$**, top** $(\ 2, 3\ )$**, side** $(\ 3, 5\ )$**.**

Example 5

Two points, $A(4, 3,\ 1\ )$ and $B(4, 3,\ 9\ )$, are projected to the screen by dropping z. Where does each land, and what does this show?

SOLUTION

1. Drop z from each: $A \to (\ 4, 3\ )$ and $B \to (\ 4, 3\ )$. *both lose z*
2. Both produce the **same** screen point even though they are at different depths.

Both land on $(\ 4, 3\ )$ **– a projection can map many 3-D points to one 2-D point, so depth is lost.**

Example 6

A point is on the screen at $(4, 3)$. List two different 3-D points that could have produced it (dropping z).

SOLUTION

1. Any 3-D point with $x = 4$ and $y = 3$ projects to $(\ 4, 3\ )$, whatever its z. *z can be anything*
2. Pick any two depths, say $z = 0$ and $z = 5$.

$(\ 4, 3, 0\ )$ **and** $(\ 4, 3, 5\ )$ **both project to** $(4, 3)$ **– there are infinitely many.**

Example 7

Project $R(\ -2\ ,\ 6\ )$ onto each axis.

SOLUTION

1. Onto the x-axis: keep $x = \ -2\ $, set $y = \ 0\ $, giving $(-2, 0)$. *keep x*
2. Onto the y-axis: keep $y = \ 6\ $, set $x = \ 0\ $, giving $(0, 6)$. *keep y*

x-axis: $(\ -2, 0\ )$; **y-axis:** $(\ 0, 6\ )$.

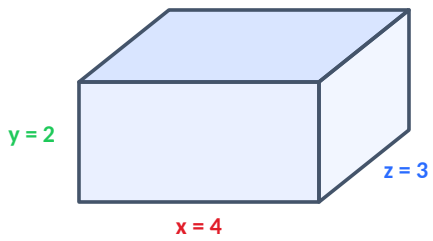
Example 8

A box sits with one corner at the origin and the opposite corner at $(4, 2, 3)$. Give the three orthographic views of that far corner (front, top, side), and state the width × height of the **front** picture.

SOLUTION

1. Front view looks along depth, dropping z: keep $(x, y) = (\ 4, 2\ )$. *drop z*
2. Top view looks down, dropping height y: keep $(x, z) = (\ 4, 3\ )$. *drop y*
3. Side view looks along width, dropping x: keep $(z, y) = (\ 3, 2\ )$. *drop x*
4. The front picture spans the kept coordinates, so its size is width 4 by height 2.

Front $(\ 4, 2\ )$, **top** $(\ 4, 3\ )$, **side** $(\ 3, 2\ )$; **the front view measures** 4×2.



The box from Example 8. The front face (x by y) is the 4×2 picture; the top drops y, the side drops x.

Example 9

A camera is positioned so the screen point is computed as $(x_s, y_s) = (x - z,\ y)$ – a simple slanted projection where depth shifts a point leftward. Project $P(\ 6\ , 4, \ 2\ )$ to the screen.

SOLUTION

1. This projection keeps y but combines x and z for the across-value: $x_s = x - z$. mix x and z
2. Across: $x_s = \ 6\ - \ 2\ = \ 4\ $.
3. Up: $y_s = \ 4\ $ (the height is carried straight through).

The screen point is $(\ 4\ ,\ 4\ )$ – depth has nudged the point 2 units left.

Example 10

Using the same rule $(x_s, y_s) = (x - z,\ y)$, two points $A(\ 6\ , 4, \ 2\ )$ and $B(\ 8\ , 4, \ 4\ )$ are drawn. Do they land on the **same** screen spot? What does this reveal?

SOLUTION

1. Project A: across $\ 6\ - \ 2\ = \ 4\ $, up 4, giving $(\ 4, 4\ )$.
2. Project B: across $\ 8\ - \ 4\ = \ 4\ $, up 4, giving $(\ 4, 4\ )$. both give 4
3. They coincide even though A and B are different 3-D points.

Both land on $(\ 4, 4\ )$ – like the drop-z projection, this slanted one is also many-to-one, so depth is still lost.

Mental-Math Shortcut

Projection is the easiest operation in geometry once you have the slogan: **"keep what you can see, zero the rest."** Onto the x-axis, the only thing you can see is how far across the point is – so keep x, make y zero. Onto a 2-D screen, you can see across and up but not depth – so keep x and y, drop z. No arithmetic at all: a projection just *deletes* a coordinate (or replaces it with 0), so you can do every example in this chapter at a glance.

Projection forces a sharp piece of logic: it is a one-way street. Knowing the original 3-D point lets you find its 2-D shadow with certainty – but knowing only the shadow does **not** let you recover the point.

Read it as an implication. *If* you know (x, y, z), *then* the screen point is exactly (x, y) – a guaranteed, single answer. The converse fails: "the screen point is $(4, 3)$, therefore the object is at ..." cannot be completed, because $(4, 3, 0)$, $(4, 3, 1)$, $(4, 3, 100)$ all satisfy it. One premise, infinitely many possible conclusions – so the reasoning is **not** reversible.

This is the precise reason you cannot judge true distance from a single photograph, and why 3-D vision and 3-D games need a **second** view (a second eye, a second camera). One projection throws away the depth; comparing **two** projections taken from different places lets you reason the depth back. The logical fix for "an implication that does not reverse" is to **add another premise** – here, a second picture.

⚠ Common Mistakes

- Thinking a projection can be undone. Dropping a coordinate destroys it: from the 2-D shadow alone you cannot recover the original 3-D point.
- Dropping the wrong coordinate for a view. The **front** view drops depth z; the **top** (bird\u2019s-eye) view drops height y. Match the discarded coordinate to the viewing direction.
- Confusing isometric with perspective. Isometric keeps equal edges equal (no shrinking); perspective makes far edges shrink towards a vanishing point. Only perspective has a horizon.
- When projecting onto an axis, moving the kept coordinate to zero instead of the other one. Onto the x-axis you keep x and zero y – the point slides straight down, not sideways.
- Forgetting that an orthographic view shows true sizes only along the directions in that view; the dropped direction is invisible in it.

Quiz

51.1.Q1. Project the 3-D point $(5, 2, 9)$ to the screen by dropping z.

A. $(5, 2)$

B. $(5, 9)$

C. $(2, 9)$

D. $(9, 2)$

51.1.Q2. Project the point $(6, 4)$ onto the x-axis.

A. $(6, 0)$

B. $(0, 4)$

C. $(0, 6)$

D. $(4, 0)$

51.1.Q3. Project the point $(6, 4)$ onto the y-axis.

51.1.Q4. Which coordinate does the **top** (bird\u2019s-eye) view drop?

A. height y

B. width x

C. depth z

D. none

51.1.Q5. True or False: from a single 2-D projection you can always recover the exact 3-D point.

A. True

B. False

51.1.Q6. In an **isometric** drawing, do equal-length edges in 3-D stay equal length on the page?

A. Yes

B. No

51.1.Q7. Both $(7, 1, 0)$ and $(7, 1, 8)$ are projected by dropping z. Do they land on the same screen point?

A. Yes

B. No

51.1.Q8. Project $(-3, 5)$ onto the x-axis.

🐍 From Concept to Code (Python)

A 3-D point is a tuple `(x, y, z)`; a screen point is `(x, y)`. Projection is the shortest function in the book – it just returns a slice of the coordinates. Graphics engines call something like this for every vertex they draw.

```
def project_to_screen(p):
    x, y, z = p
    return (x, y)          # drop the depth z

def project_to_x_axis(p):
    x, y = p
    return (x, 0)          # keep x, zero y

print(project_to_screen((4, 3, 7)))
print(project_to_x_axis((4, 3)))
```

```
output
(4, 3)
(4, 0)
```

```
# Many 3-D points collapse to one screen point
depths = [0, 1, 5, 20]
for z in depths:
    print((4, 3, z), "->", project_to_screen((4, 3, z)))
```

```
output
(4, 3, 0) -> (4, 3)
(4, 3, 1) -> (4, 3)
(4, 3, 5) -> (4, 3)
(4, 3, 20) -> (4, 3)
```

Try it:

1. Write `project_to_y_axis(p)` that keeps the y-value and zeroes x, then test it on `(6, 4)`.
2. Write `top_view(p)` for a 3-D point that drops the height y (keep x and z). What does `top_view((2, 5, 3))` give?

Answer Key – Chapter 51 – Projections

51.1.Q1. $(5, 2)$ – keep x and y, discard the depth $z = 9$.

51.1.Q2. $(6, 0)$ – keep the x-value, set y to 0 (drop straight down).

51.1.Q3. $(0, 4)$ – keep the y-value, set x to 0.

51.1.Q4. Height y – looking straight down keeps (x, z) and removes the height.

51.1.Q5. False – the dropped coordinate is lost; many 3-D points share one shadow.

51.1.Q6. Yes – isometric preserves equal edge lengths; it is perspective that shrinks distant edges.

51.1.Q7. Yes – both give $(7, 1)$; depth information is lost in the projection.

51.1.Q8. $(-3, 0)$ – keep $x = -3$, set $y = 0$.

Chapter 52 · Intersection & Collision Geometry

The handful of yes/no tests that every game and graphics program uses to decide whether two things touch: is a point inside a rectangle or a circle, do two boxes overlap (AABB), and have two circles collided – each reduced to simple arithmetic and a logical AND.

The Idea Behind It

Two circles overlap exactly when the distance d between their centres satisfies $d \le r_1 + r_2$.

but why?

Draw the straight line joining the two centres C_1 and C_2. The closest a point of circle 1 can ever get to circle 2 is reached *along this line*, at the rim point that faces C_2 – sticking out a distance r_1 from C_1. The same is true the other way, so the question "do the circles touch?" is really "do these two reaching arms meet in the middle?" The governing fact is just how the lengths add up along that one line:

$$\boxed{\text{gap between the rims} = d - (r_1 + r_2)}$$

Start from the full centre-to-centre distance d, then spend r_1 getting from C_1 out to its rim, and r_2 getting from C_2 out to its rim. What is **left over** is the clear space between the two near edges:

$$\text{gap} = d - r_1 - r_2 = d - (r_1 + r_2)$$

The circles fail to meet only while that leftover gap is positive. They first **touch** when it closes to zero, and **overlap** once it goes negative:

$$d - (r_1 + r_2) \le 0 \iff d \le r_1 + r_2$$

$$\therefore \text{ the circles overlap} \iff d \le r_1 + r_2$$

So one comparison settles it: measure the centre gap d, compare it with the combined reach $r_1 + r_2$, and you never need the circles' shapes at all.

touch at equality $d = r_1 + r_2$; strict overlap when $d < r_1 + r_2$

When two spaceships crash, a bullet hits a wall, or a mouse-click lands on a button, a computer has to answer one blunt question: *are these two shapes touching?* These are **collision tests**, and they run thousands of times a second. The beautiful part is that each one boils down to a comparison of numbers joined by the word **AND** – the logic you met in Part V, now doing real work.

Is a point inside a rectangle?

We describe an upright rectangle by the range it covers on each axis: from $x_{\min}$ to $x_{\max}$ across, and $y_{\min}$ to $y_{\max}$ up. A point (p_x, p_y) is **inside** only if it lies within the across-range *and* within the up-range:

$$(x_{\min} \le p_x \le x_{\max}) \ \textbf{AND} \ (y_{\min} \le p_y \le y_{\max})$$

Both halves must be true. If the point is in the right column of space but too high, the second half fails and the whole test fails – that is the **AND** at work.

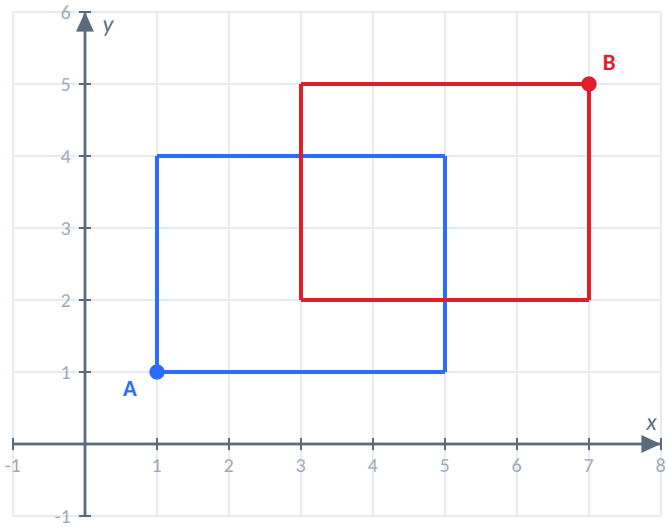
Do two boxes overlap? (AABB)

> **AABB**
>
> An **AABB** (axis-aligned bounding box) is an upright rectangle, lined up with the axes, used as a stand-in for a more complicated shape. Most games wrap each object in an AABB because two AABBs are very cheap to test.

AABB collision bounding circle

Two boxes overlap when their shadows overlap on **both** axes at once. Think of each box's range on the x-axis and on the y-axis. The boxes touch only if the x-ranges overlap **AND** the y-ranges overlap. Two number-ranges $[a_1, a_2]$ and $[b_1, b_2]$ overlap exactly when *each one starts before the other ends*:

$$\underbrace{(A_{x\,\min} \le B_{x\,\max} \textbf{ and } B_{x\,\min} \le A_{x\,\max})}_{x\text{-ranges overlap}} \textbf{ AND } \underbrace{(A_{y\,\min} \le B_{y\,\max} \textbf{ and } B_{y\,\min} \le A_{y\,\max})}_{y\text{-ranges overlap}}$$



Two AABBs that overlap. Their x-ranges (1–5 and 3–7) share 3–5, AND their y-ranges (1–4 and 2–5) share 2–4, so they collide.

Is a point inside a circle? Have two circles collided?

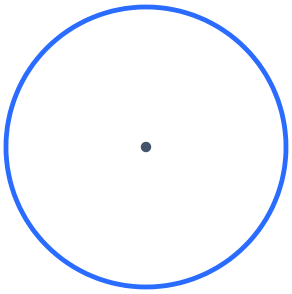
Circles are even friendlier, because a circle is "every point within distance r of the centre". The straight-line distance between two points (x_1, y_1) and (x_2, y_2) comes from Pythagoras:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

A point is **inside** a circle of centre C and radius r when its distance to the centre is at most r: $d \le r$.

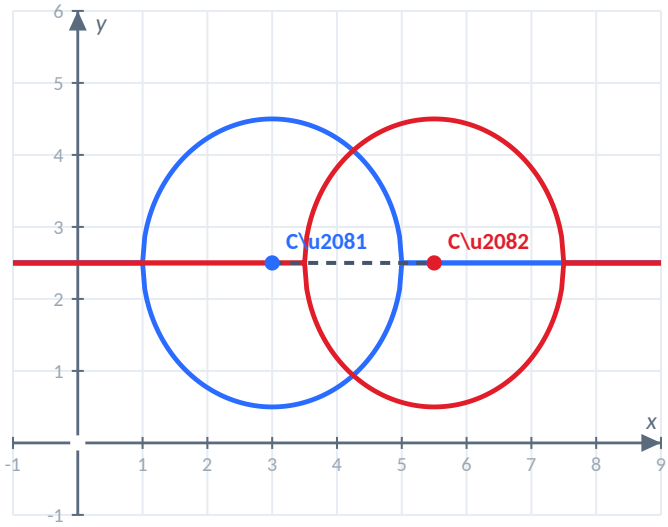
Two circles **collide** when they are close enough that they touch or overlap – that is, when the distance between their centres is **less than the sum of their radii**:

$$d < r_1 + r_2 \quad \Longrightarrow \quad \text{the circles overlap}$$



Two overlapping circles. If the gap between centres is smaller than $r_1 + r_2$, they intersect – the single cleanest collision test in all of graphics.

Here is the picture in coordinates: two circles whose centres are close, drawn as a plot, so you can see the overlap region where they meet.



Centres 2.5 apart, each radius 2, so $r_1+r_2 = 4$. Since 2.5 < 4, the circles overlap (they share the lens-shaped middle).

Note: A neat speed trick: comparing $d < r_1 + r_2$ needs a square root, which is slow. Square both sides instead and compare $d^2 < (r_1 + r_2)^2$ – same answer, no square root. Games use this constantly.

Worked examples

Example 1

A rectangle spans x from 0 to 5 and y from 0 to 3. Is the point $(4, 2)$ inside it?

SOLUTION

1. Check the across-range: is $0 \leq 4 \leq 5$? Yes.
2. Check the up-range: is $0 \leq 2 \leq 3$? Yes.
3. Inside requires **both** to be true (an AND); both hold. *both true → in*

Yes – $(4, 2)$ is inside the rectangle.

Example 2

Same rectangle (x: 0–5, y: 0–3). Is the point $(4, 7)$ inside?

SOLUTION

1. Across-range: is $0 \leq 4 \leq 5$? Yes.
2. Up-range: is $0 \leq 7 \leq 3$? No – 7 is above the top. *7 > 3*
3. AND needs both true; the second is false, so the AND is false.

No – the point is too high, so it lies outside.

Example 3

Box A covers $x \in [1, 5], y \in [1, 4]$. Box B covers $x \in [3, 7], y \in [2, 5]$. Do they overlap?

SOLUTION

1. x-ranges overlap? A starts at $1 \leq 7$ (B\u2019s max) and B starts at $3 \leq 5$ (A\u2019s max). Yes.
2. y-ranges overlap? A starts at $1 \leq 5$ and B starts at $2 \leq 4$. Yes.
3. Overlap needs both axes to overlap (AND); both do. *x and y both share*

Yes – the boxes collide (this is the figure above).

Example 4

Box A covers $x \in [0, 2], y \in [0, 2]$. Box B covers $x \in [5, 7], y \in [0, 2]$. Do they overlap?

SOLUTION

1. x-ranges overlap? A\u2019s max is 2, B\u2019s min is 5. Since $2 < 5$ they do not meet on x.
   gap on x
2. The y-ranges happen to match, but the x test already failed.
3. The AND is false the moment one axis fails.

No – there is a horizontal gap, so the boxes do not collide.

Example 5

Is the point $(5, 6)$ inside a circle centred at $(2, 2)$ with radius $r = 5$?

SOLUTION

1. Find the distance from the point to the centre: differences are $5 - 2 = 3$ and $6 - 2 = 4$.
2. $d = \sqrt{3^2 + 4^2} = \sqrt{9 + 16} = \sqrt{25} = 5$. 3-4-5 triangle
3. Inside (or on the edge) means $d \leq r$: here $5 \leq 5$.

Yes – it sits exactly on the boundary, which counts as inside.

Example 6

Circle 1: centre $(0, 0)$, radius 3. Circle 2: centre $(4, 0)$, radius 2. Have they collided?

SOLUTION

1. Distance between centres: differences $4 - 0 = 4$ and $0 - 0 = 0$, so $d = \sqrt{4^2 + 0^2} = 4$.
2. Sum of radii: $r_1 + r_2 = 3 + 2 = 5$.
3. Collision test: is $d < r_1 + r_2$? Is $4 < 5$? Yes. d < r1+r2

Yes – the circles overlap, since $4 < 5$.

Example 7

Circle 1: centre $(0, 0)$, radius 2. Circle 2: centre $(10, 0)$, radius 3. Have they collided?

SOLUTION

1. Distance between centres: $d = \sqrt{10^2 + 0^2} = 10$.
2. Sum of radii: $2 + 3 = 5$.
3. Is $10 < 5$? No. too far apart

No – the centres are far apart ($10 > 5$), so the circles do not touch.

Example 8

Two circles have centres $(1, 2)$ and $(4, 6)$, with radii $r_1 = 3$ and $r_2 = 4$. Use the squared-distance speed trick to decide if they collide.

SOLUTION

1. Differences between centres: $\Delta x = 4 - 1 = 3$, $\Delta y = 6 - 2 = 4$.
2. Squared distance (no root needed): $d^2 = 3^2 + 4^2 = 9 + 16 = 25$. skip the √
3. Squared reach: $(r_1 + r_2)^2 = (3 + 4)^2 = 7^2 = 49$.
4. Collision test on the squares: is $25 < 49$? Yes. compare the squares

Yes – since $25 < 49$ the circles overlap, and we never took a square root.

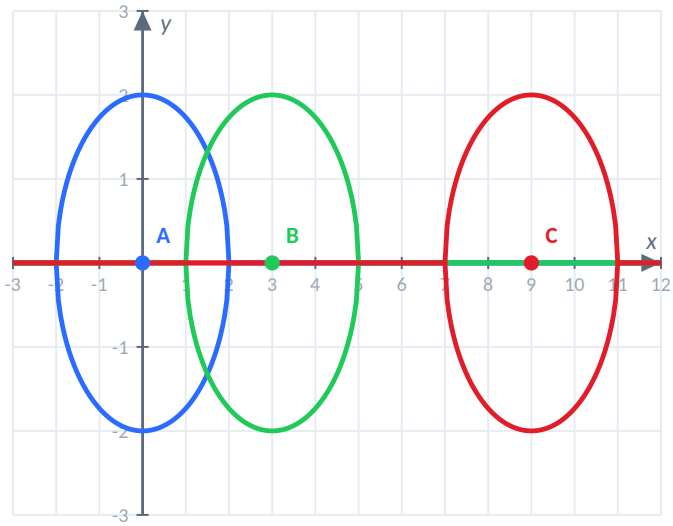
Example 9

Three circles A, B, C lie in a row: A at $(0, 0)$ with $r = 2$, B at $(3, 0)$ with $r = 2$, C at $(9, 0)$ with $r = 2$. Which pairs collide?

SOLUTION

1. A–B: $d = \sqrt{3^2 + 0^2} = 3$; sum of radii $2 + 2 = 4$. Is $3 < 4$? Yes – they touch.
2. B–C: $d = \sqrt{6^2 + 0^2} = 6$; sum 4. Is $6 < 4$? No – a gap. 6 > 4
3. A–C: $d = \sqrt{9^2 + 0^2} = 9$; sum 4. Is $9 < 4$? No – further still.

Only A and B collide; B–C and A–C are too far apart. The plot below shows the chain.



A (blue) and B (green) overlap since their centres are only 3 apart (< 4). C (red) sits well clear of both.

Example 10

A circular character of radius 1 is centred at $(6, 2)$. A solid wall fills the rectangle $x \in [0, 5]$, $y \in [0, 4]$. Has the character hit the wall? (Test the closest point of the rectangle to the centre.)

SOLUTION

1. Clamp the centre into the box to find the nearest wall point: clamp $x = 6$ into $[0, 5]$ gives 5; clamp $y = 2$ into $[0, 4]$ gives 2. nearest point = (5, 2)
2. Distance from the centre $(6, 2)$ to that nearest point $(5, 2)$: $d = \sqrt{(6 - 5)^2 + (2 - 2)^2} = \sqrt{1} = 1$.
3. A hit means this distance is at most the radius: is $1 \leq 1$? Yes (just touching). $d \leq r$

Yes – the nearest wall point is exactly 1 unit away, equal to the radius, so the circle grazes the wall.

⚡ Mental-Math Shortcut

The "separated on any axis" shortcut for boxes. Rather than checking the four overlap conditions, ask the *opposite* question, which is faster to see: are the boxes **separated**? Two boxes miss each other if one is entirely to the left of the other ($A_{x\,\max} < B_{x\,\min}$) **OR** entirely to the right **OR** entirely above **OR** entirely below. If *none* of those four "clear gap" situations is true, they must overlap. Spotting a single clear gap on one axis instantly tells you "no collision" without any further work.

For **circles**, skip the square root: compare d^2 with $(r_1 + r_2)^2$. Squaring two small numbers in your head beats taking a root every time.

A collision test is a worked example of **logical AND** (Chapter 18) and its mirror image under **De Morgan's law** (Chapter 21). Take two boxes overlapping. "They collide" is the conjunction

$$\text{collide} \;=\; (\text{overlap on } x)\ \mathbf{AND}\ (\text{overlap on } y).$$

For an AND to be **true, both** parts must be true. So a *single* axis with a clear gap makes the whole thing false – which is why checking can stop early the instant one axis separates.

Now negate it. "They do **not** collide" is $\neg(\,P \wedge Q\,)$, and De Morgan turns that into $\neg P\ \mathbf{OR}\ \neg Q$:

$$\text{no collision} \;=\; (\text{a gap on } x)\ \mathbf{OR}\ (\text{a gap on } y).$$

Read it aloud: the boxes miss each other if there is a gap on x *or* a gap on y – exactly the mental-math shortcut above, and exactly what De Morgan predicts. The same structure scales to 3-D boxes: collision is an AND over **three** axes, and a gap on *any one* of them means "no collision". One logical rule, ANDed across however many dimensions you have.

⚠ Common Mistakes

- Using OR where AND is needed. A point is inside a rectangle only if it is in range on **both** axes; "in range on x **or** y" is wrong and would mark far-away points as inside.
- Forgetting an axis in the AABB test. Boxes overlap only when **both** the x-ranges and the y-ranges overlap; overlapping on one axis alone is not a collision.
- Comparing the distance with one radius instead of the **sum** of radii. Two circles collide when $d < r_1 + r_2$, not when $d < r_1$.
- Forgetting to square the differences in the distance formula. It is $\sqrt{(\Delta x)^2 + (\Delta y)^2}$ – you must square *before* adding, then take the root.
- Squaring only one side of the speed-up. If you drop the square root on the distance, you must compare with $(r_1 + r_2)^2$, the **squared** sum, not $r_1 + r_2$.

Quiz

52.1.Q1. A rectangle spans $x \in [0, 5]$, $y \in [0, 3]$. Is $(2, 1)$ inside?

A. Yes

B. No

52.1.Q2. Same rectangle. Is $(2, 9)$ inside?

A. Yes

B. No

52.1.Q3. Two AABBs overlap only when their ranges overlap on how many axes?

A. one

B. either

C. both

D. none

52.1.Q4. Circles collide when the distance between centres is …

A. $> r_1 + r_2$

B. $< r_1 + r_2$

C. $< r_1$

D. $= 0$

52.1.Q5. Find the distance between $(0, 0)$ and $(6, 8)$.

52.1.Q6. Circle 1: centre $(0, 0)$, $r = 3$. Circle 2: centre $(4, 0)$, $r = 2$. Collision?

A. Yes

B. No

52.1.Q7. Is the point $(3, 4)$ inside a circle centred at the origin with radius 5?

52.1.Q8. True or False: to avoid a slow square root, you can compare d^2 with $(r_1 + r_2)^2$.

A. True

B. False

🐍 From Concept to Code (Python)

Each test is a function that returns `True` or `False`. The AABB test is a single chained `and`; the circle test uses the squared-distance trick to skip the square root. This is essentially the real code inside a 2-D game engine.

```
def aabb_overlap(ax1, ay1, ax2, ay2, bx1, by1, bx2, by2):
    return (ax1 <= bx2 and bx1 <= ax2 and   # x-ranges overlap
            ay1 <= by2 and by1 <= ay2)        # AND y-ranges overlap

# Box A = (1,1)-(5,4),  Box B = (3,2)-(7,5)
print(aabb_overlap(1, 1, 5, 4, 3, 2, 7, 5))
# Box A = (0,0)-(2,2),  Box C = (5,0)-(7,2)
print(aabb_overlap(0, 0, 2, 2, 5, 0, 7, 2))
```

```
output
True
False
```

```
def circles_collide(x1, y1, r1, x2, y2, r2):
    dx, dy = x2 - x1, y2 - y1
    dist_sq = dx * dx + dy * dy        # squared distance, no sqrt
    reach = r1 + r2
    return dist_sq < reach * reach     # compare squared values

print(circles_collide(0, 0, 3, 4, 0, 2))    # 4 < 5 -> touch
print(circles_collide(0, 0, 2, 10, 0, 3))   # 10 > 5 -> miss
```

```
output
True
False
```

Try it:

1. Write `point_in_rect(px, py, x1, y1, x2, y2)` returning `True` when the point is inside; test `(4, 2)` in the box `(0,0)-(5,3)`.
2. Write `point_in_circle(px, py, cx, cy, r)` using the squared-distance idea, and check that `(3, 4)` is inside the circle at the origin with `r = 5`.

Answer Key – Chapter 52 – Intersection & Collision Geometry

52.1.Q1. Yes – $0 \leq 2 \leq 5$ **and** $0 \leq 1 \leq 3$ are both true.

52.1.Q2. No – 9 is outside the y-range $[0, 3]$, so the AND fails.

52.1.Q3. Both – the x-ranges **and** the y-ranges must overlap (a logical AND).

52.1.Q4. $< r_1 + r_2$ – closer than the sum of the radii means they overlap.

52.1.Q5. 10 – $\sqrt{6^2 + 8^2} = \sqrt{36 + 64} = \sqrt{100} = 10$.

52.1.Q6. Yes – centres are 4 apart and $r_1 + r_2 = 5$; since $4 < 5$ they overlap.

52.1.Q7. Yes – its distance is $\sqrt{3^2 + 4^2} = 5$, and $5 \leq 5$ (on the boundary, which counts).

52.1.Q8. True – squaring both sides preserves the comparison and skips the square root.

Chapter 53 · Graphs & Networks

> *How to model anything made of "things and the connections between them" – friends, roads, web links, computers – as a graph of nodes joined by edges, and how to count connections, follow paths, and answer "can I get from here to there?"*

The Idea Behind It

Add up the degrees of every node and you always get twice the number of edges: $\sum \deg(v) = 2E$.

but why?

The degree of a node counts the edge-ends sticking out of it. Adding all the degrees therefore counts *every edge-end in the whole graph*. So instead of asking "how many edges?", ask the easier question "how many edge-ends?" The one fact that cracks it is almost too simple to notice, **every edge has exactly two endpoints**:

$$\boxed{\text{each edge contributes 2 ends (one at each of its two nodes)}}$$

Count the total a second way, edge by edge. Each of the E edges plants one end in each of its two nodes, adding 2 to the running degree total. Sum over all the edges:

$$\sum_{v} \deg(v) = \underbrace{2 + 2 + \cdots + 2}_{E \text{ edges}} = 2E$$

$$\therefore \sum_{v} \deg(v) = 2E$$

Because the total is $2E$, it is always **even** – and that forces the count of odd-degree nodes to be even too (the odd ones must pair up to keep the sum even). This is the **handshaking lemma**: in any party, the number of people who shook an odd number of hands is even.

a handy self-check: if your degrees sum to an odd number, you miscounted an edge

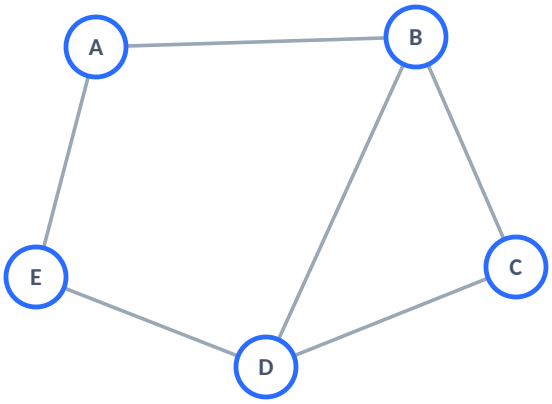
A road map, a friend list, the internet, a family tree, the London Underground – all of these are the same kind of object underneath. Each is a set of **things** with **connections** between some pairs of them. Mathematicians call that object a **graph** (nothing to do with the graphs-of-functions kind), and it is one of the most useful ideas in all of computing.

GRAPH

A **graph** is a set of **nodes** (also called *vertices* – the "things") together with a set of **edges** (the connections, each joining two nodes). We draw nodes as circles and edges as lines between them.

node edge degree path

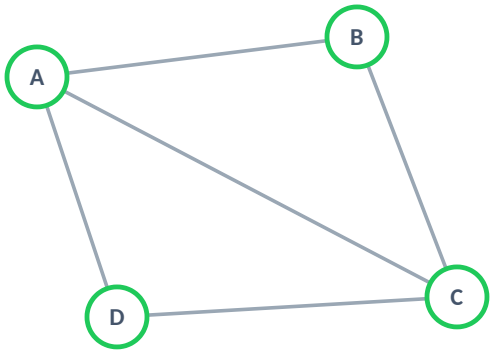
Here is a small friendship graph. Five people are the nodes; an edge means "these two are friends".



A friendship graph: 5 nodes, 6 edges. An edge between two circles means they are connected (friends).

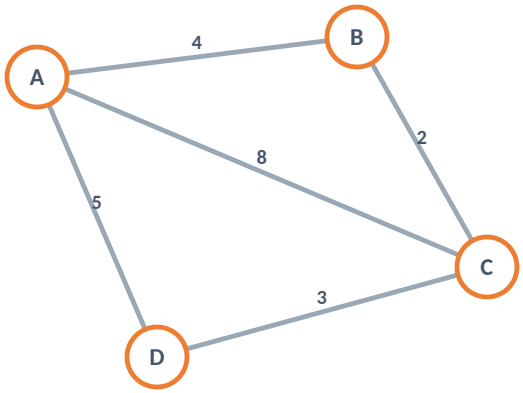
Directed and weighted graphs

Sometimes a connection only goes **one way**. "A follows B" on social media does not mean B follows A. A graph whose edges have arrows is **directed**; one with plain lines (the friendship kind, where the link is mutual) is **undirected**.



A directed graph: the arrows show one-way links, e.g. A → B (A points to B, not the reverse).

Often each edge also carries a **weight** – a number such as a distance, a cost, or a travel time. A road map is a **weighted** graph: nodes are towns, edge weights are the miles between them.



A weighted road graph: each edge is labelled with its distance. The direct A–C road (8) is longer than going A–B–C (4+2 = 6).

Degree – counting connections

The **degree** of a node is simply how many edges touch it. In the friendship graph above, node D is joined to B, C and E, so its degree is 3 – D has three friends. Adding up the degrees of every node always gives **twice** the number of edges, because each edge has two ends and so is counted at both of them.

Paths and reachability

A **path** is a route that hops from node to node along edges. In the friendship graph you can get from A to C by the path $A \to B \to C$. If *some* path exists between two nodes, the second is **reachable** from the first. If every node can reach every other, the graph is **connected** – all in one piece, with no island cut off from the rest.

> **Note:** Two nodes can be reachable without a direct edge. There is no single edge from A to C in the friendship graph, yet C is reachable from A – through B (or through E then D). Reachability asks whether *any* route exists, however many hops it takes.

Storing a graph: the adjacency list

To put a graph into a computer, the simplest method is an **adjacency list**: for each node, keep the list of its neighbours. The friendship graph becomes:

- A: B, E
- B: A, C, D
- C: B, D
- D: B, C, E
- E: A, D

Reading off a node's degree is now instant – just count the length of its list. (In Python this is a dictionary, as the code section shows.)

Worked examples

Example 1

In the friendship graph (A:B, E · B:A, C, D · C:B, D · D:B, C, E · E:A, D), what is the degree of node B?

SOLUTION

1. Degree is the number of edges at the node – the length of its neighbour list. *count the list*
2. B is joined to A, C, D, so it has 3 neighbours.

Degree of B is 3.

Example 2

Find the degree of every node, then check the handshake rule.

SOLUTION

1. Count each list: A has 2, B has 3, C has 2, D has 3, E has 2.
2. Add the degrees: $2 + 3 + 2 + 3 + 2 = 12$.
3. The graph has 6 edges, and $2 \times 6 = 12$. They match (each edge counted at both ends).

sum = 2 × edges

Degrees $2, 3, 2, 3, 2$; their sum $12 = 2 \times 6$ edges.

Example 3

Is node C reachable from node A? Give a path.

SOLUTION

1. Look for a route hopping along edges. From A you can reach B (they are neighbours).
2. From B you can reach C (also neighbours). *hop along edges*
3. So $A \to B \to C$ is a valid path.

Yes – e.g. the path $A \to B \to C$ (length 2).

Example 4

Find a different path from A to C that does not use the edge B–C.

SOLUTION

1. Start at A; its neighbours are B and E. Try E to avoid going through B to C. *dodge B–C*
2. From E go to D (neighbours).
3. From D go to C (neighbours).

The path $A \to E \to D \to C$ reaches C without the B–C edge.

Example 5

In the directed graph with edges $A \to B$, $B \to C$, $C \to D$, $D \to A$, $A \to C$: can you travel from A to D? Can you travel from D to B?

SOLUTION

1. Edges are one-way – you may only move along an arrow. *follow the arrows*
2. A to D: take $A \to C$ then $C \to D$. Yes.
3. D to B: from D the only arrow is $D \to A$; from A go $A \to B$. Yes, via $D \to A \to B$.

Both are possible: $A \to C \to D$, and $D \to A \to B$.

Example 6

In the weighted road graph (A–B:4, B–C:2, A–D:5, D–C:3, A–C:8), which route from A to C is shortest?

SOLUTION

1. List the routes and add their edge weights.
2. Direct A–C: 8.
3. Via B: A–B + B–C = $4 + 2 = 6$. *add the weights*
4. Via D: A–D + D–C = $5 + 3 = 8$.
5. Compare 8, 6, 8 – the smallest is 6.

The route $A \to B \to C$, total distance 6 (shorter than the direct road).

Example 7

A graph has nodes $\{P, Q, R, S\}$ and only the edges P–Q and R–S. Is it connected?

SOLUTION

1. Connected means every node can reach every other by some path.
2. P can reach Q, and R can reach S – but there is no edge linking the $\{P, Q\}$ pair to the $\{R, S\}$ pair.
   two islands
3. So P cannot reach R at all.

No – it splits into two separate pieces $\{P, Q\}$ and $\{R, S\}$; it is not connected.

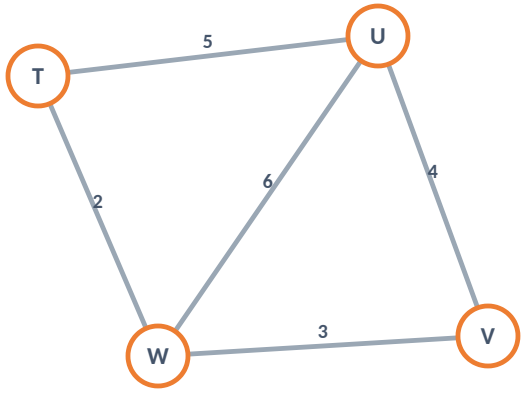
Example 8

A delivery network has towns T, U, V, W with roads T–U:5, U–V:4, T–W:2, W–V:3, U–W:6. Find the shortest route from T to V.

SOLUTION

1. List every simple route from T to V and add the weights along it.
2. Via U: T–U + U–V = $5 + 4 = 9$.
3. Via W: T–W + W–V = $2 + 3 = 5$. *add the weights*
4. Via W then U: T–W + W–U + U–V = $2 + 6 + 4 = 12$.
5. Compare 9, 5, 12 – the smallest total is 5.

The shortest route is $T \to W \to V$, total distance 5.



The delivery network of Example 8. The route T–W–V (2+3 = 5) beats the direct-feeling T–U–V (5+4 = 9).

Example 9

In the directed graph with edges $A \to B$, $B \to C$, $C \to A$ (a 3-cycle) plus a one-way $C \to D$, list every node reachable **from** A, and decide whether A is reachable **from** D.

SOLUTION

1. Spread from A along the arrows only. Start $\{A\}$; $A \to B$ adds B.
2. From B: $B \to C$ adds C. From C: $C \to A$ (already in) and $C \to D$ adds D. *follow arrows*
3. Nothing new remains, so from A you can reach B, C, D (and A itself).
4. Now from D: D has no outgoing arrows, so you are stuck at D. *dead end*

From A: B, C, D are all reachable. But A is not reachable from D – D is a one-way sink.

Example 10

A graph has 5 nodes. Four of them have degrees $3, 3, 2, 4$. The degree total must be even (handshake rule); if the graph has 8 edges, what is the degree of the fifth node?

SOLUTION

1. The handshake rule says the degrees sum to twice the edges: $2 \times 8 = 16$. *sum = 2 × edges*
2. Add the four known degrees: $3 + 3 + 2 + 4 = 12$.
3. The fifth degree fills the gap: $16 - 12 = 4$.

The fifth node has degree 4 (and the total $3 + 3 + 2 + 4 + 4 = 16 = 2 \times 8$ checks out).

Mental-Math Shortcut

The handshake check. Whenever you list the degrees of all the nodes, their **total is always even** – it equals twice the number of edges, because every edge is counted at both of its ends. So if you add up your degrees and get an *odd* number, you have made a mistake somewhere (a miscounted or one-ended edge). It is the fastest sanity check in graph theory: *degrees must sum to an even number*, and that even number divided by 2 is the edge count.

"Is B reachable from A?" looks like a single yes/no question, but answering it is a piece of step-by-step logical reasoning – and it is exactly how a computer explores a maze or a network.

Define a set R of nodes "known to be reachable from A". The reasoning is a rule applied over and over:

$$\text{Start: } A \in R. \qquad \text{Rule: if } x \in R \text{ and there is an edge } x - y, \text{ then } y \in R.$$

Read the rule as an implication: *being reachable spreads along edges*. Keep applying it until nothing new can be added. Then conclude with a clean piece of logic:

$$B \text{ is reachable} \iff B \in R \text{ when the spreading stops.}$$

Trace it on the friendship graph from A: start $\{A\}$; A's edges add B, E; from B add C, D; from E, D is already in. Nothing new remains, so $R = \{A, B, C, D, E\}$ – *every* node, so the graph is connected.

Notice the shape of the argument: a **base case** (A is reachable) plus a **spreading rule** that fires while it can. That is the same "start + rule" engine as a recurrence (Chapter 33) and as mathematical induction – and "reachable means: reached by repeatedly following the rule" is how the question gets a definite answer.

⚠ Common Mistakes

- Confusing this kind of graph with the graph of a function. Here a "graph" is **nodes joined by edges** (a network), not a curve on axes.
- Treating a directed edge as two-way. An arrow $A \to B$ lets you go from A to B only; you cannot travel back along it unless a separate edge $B \to A$ exists.
- Miscounting degree by missing an edge. The degree of a node is **every** edge touching it; in an adjacency list it is the full length of its neighbour list.
- Assuming two nodes need a direct edge to be reachable. Reachability allows **any** number of hops – a multi-step path counts.
- Thinking a graph with some edges must be connected. If even one node (or group) has no route to the rest, the graph is disconnected.

Quiz

53.1.Q1. In a graph, what is an **edge**?

A. a connection between two nodes

B. a number on a node

C. a curve on axes

D. the centre of a circle

53.1.Q2. A node is joined to exactly 4 others. What is its degree?

A. 2

B. 4

C. 8

D. 1

53.1.Q3. A graph has 5 edges. What do all the node degrees add up to?

A. 5

B. 10

C. 25

D. 2

53.1.Q4. In a **directed** edge $A \to B$, can you travel from B to A along it?

A. Yes

B. No

53.1.Q5. What does it mean that node C is **reachable** from node A?

53.1.Q6. A weighted graph has A–B = 4 and B–C = 2. How long is the path $A \to B \to C$?

53.1.Q7. True or False: a graph with at least one edge is always connected.

A. True

B. False

53.1.Q8. In an adjacency list, the entry for node D is $[B, C, E]$. What is D\u2019s degree?

🐍 From Concept to Code (Python)

The natural way to hold a graph in Python is a **dictionary**: each node maps to a list of its neighbours. From that one structure you can read off degrees, list neighbours, and walk paths – exactly the adjacency list from the lesson.

```
graph = {
    "A": ["B", "E"],
    "B": ["A", "C", "D"],
    "C": ["B", "D"],
    "D": ["B", "C", "E"],
    "E": ["A", "D"],
}

print(graph["D"])          # D's neighbours
print(len(graph["B"]))      # degree of B
print(sum(len(v) for v in graph.values()))  # 2 x edges
```

```
output
['B', 'C', 'D']
3
12
```

```
# Is C reachable from A? Spread along edges until nothing new appears.
def reachable(graph, start):
    seen, stack = {start}, [start]
    while stack:
        node = stack.pop()
        for nb in graph[node]:
            if nb not in seen:
                seen.add(nb)
                stack.append(nb)
    return seen

print("C" in reachable(graph, "A"))   # can A reach C?
```

```
output
True
```

Try it:

1. Print the degree of every node by looping over `graph` and using `len(neighbours)`.
2. Add a new friend `"F"` connected only to `"E"`, then check whether `reachable(graph, "A")` now includes `"F"`.

Answer Key – Chapter 53 – Graphs & Networks

53.1.Q1. A connection between two nodes – the line joining two circles.

53.1.Q2. 4 – degree is the number of edges meeting at the node.

53.1.Q3. 10 – the degree total is always twice the edge count, $2 \times 5 = 10$.

53.1.Q4. No – a directed edge is one-way; you would need a separate $B \to A$ edge.

53.1.Q5. Some path (one or more edges) leads from A to C – a direct edge is not required.

53.1.Q6. 6 – add the edge weights along the path: $4 + 2 = 6$.

53.1.Q7. False – it is connected only if **every** node can reach every other; isolated pieces make it disconnected.

53.1.Q8. 3 – the degree is the length of its neighbour list, here 3.



PART X

Calculus

The mathematics of change. How to measure an instant of motion, the slope of a curve, the area beneath it, and how everything from optimisation to prediction grows out of two ideas: the derivative and the integral.

Chapter 54 – Limits · Chapter 55 – Derivatives · Chapter 56 – Rates of Change & Motion · Chapter 57 – Optimisation · Chapter 58 – Integrals · Chapter 59 – Multivariable Calculus: Partial Derivatives & Gradients · Chapter 60 – Differential Equations · Chapter 61 – Series & Approximations

Chapter 54 · Limits

> *What it means for a function to **approach** a value as the input creeps toward a point (even at a spot where the formula itself breaks down), and how this single idea quietly powers all of calculus.*

The Idea Behind It

A function can head straight for a value at a point where it is undefined: $\frac{x^2-1}{x-1} \to 2$ as $x \to 1$, even though plugging in $x = 1$ gives $\frac{0}{0}$. *but why?*

A limit asks where a curve is *going*, not where it lands. So a single missing point cannot block it, as long as the curve still aims at one clear height from both sides. The trick is to rewrite the formula in a way that survives at the hole, using a rule you already trust: a common factor on top and bottom **cancels** wherever it is non-zero.

$$\boxed{\frac{(x-1)\,g(x)}{(x-1)} = g(x) \quad \text{for every } x \neq 1}$$

Factor the top as a difference of squares, $x^2 - 1 = (x-1)(x+1)$, then cancel the troublesome $x-1$ that was causing the $\frac{0}{0}$:

$$\frac{x^2-1}{x-1} = \frac{(x-1)\,(x+1)}{(x-1)} = x+1 \quad (x \neq 1)$$

Away from the hole the function is simply the line $x+1$, which is perfectly well behaved. As x creeps toward 1, that line heads for $1+1 = 2$ from the left and the right alike:

$$\lim_{x\to 1} \frac{x^2-1}{x-1} = \lim_{x\to 1}(x+1) = 2$$

$$\therefore \text{ the limit is } 2, \text{ a value the function itself never takes.}$$

So $\frac{0}{0}$ is never the final word, it is a signal to factor and cancel, then look again. The limit lives in the *journey*, so the one absent point at the end cannot spoil it. *destination, not the missing step*

Sometimes the interesting question about a function is not "what is its value *here*?" but "what value is it *heading toward* as we get near here?" Those are different questions, and the gap between them is the whole point of a **limit**. A limit watches where a curve is going, not where it lands.

Sneaking up on a value

Picture walking toward a doorway. With each step you halve the remaining distance. You get closer and closer to the door, settling in on it without ever needing to talk about the door itself. A limit captures exactly that "settling in". We write

$$\lim_{x\to a} f(x) = L$$

and read it as: *as x approaches a, the value of $f(x)$ approaches L.* The arrow $\to$ means "gets ever closer to". Crucially, x never has to actually equal a – we only care about the journey, not the destination.

LIMIT

The **limit** of $f(x)$ as $x \to a$ is the single value L that $f(x)$ settles toward as x is taken closer and closer to a (from both sides). It is written $\lim_{x \to a} f(x) = L$.

Limit

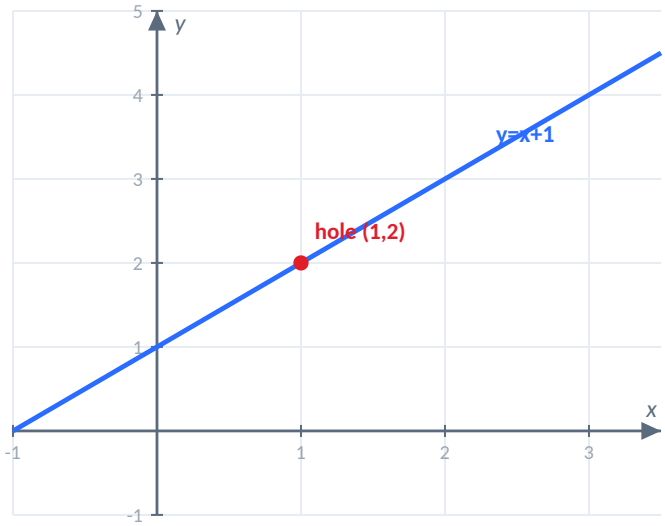
When the formula has a hole

Limits earn their keep when the formula itself stumbles. Consider

$$f(x) = \frac{x^2 - 1}{x - 1}.$$

At $x = 1$ this is $\frac{0}{0}$ – undefined, a genuine blank spot. But *near* $x = 1$ the function behaves perfectly. The numerator factors as $x^2 - 1 = (x - 1)(x + 1)$, so for every x except 1 the $(x - 1)$ cancels and $f(x) = x + 1$. As x creeps toward 1, the value creeps toward $1 + 1 = 2$. The graph is just the line $y = x + 1$ with a single point punched out at $(1, 2)$ – a **removable hole**.

Removable hole



f(x)=(x²−1)/(x−1) is the line y=x+1 with one point removed at x=1. The curve still aims straight at the red hole, so the limit is 2.

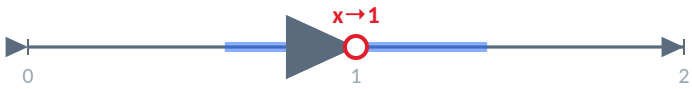
The open red dot marks the missing point. Notice the curve still *aims* squarely at it from both sides. The function has no value at $x = 1$, yet it has a perfectly clear limit there: **2**. Value and limit are separate ideas.

Watching it settle with a table

If algebra feels abstract, just plug in numbers that hug $x = 1$ and read off the trend. Approaching from below ($x < 1$) and from above ($x > 1$):

x	$f(x) = \dfrac{x^2 - 1}{x - 1}$
0.9	1.9
0.99	1.99
0.999	1.999
$\rightarrow 1 \leftarrow$	$\rightarrow 2 \leftarrow$
1.001	2.001
1.01	2.01
1.1	2.1

From either direction the output is squeezed toward 2. That agreement from both sides is what lets us declare the limit equals 2 with confidence. Picture the inputs closing in on 1 on the number line below – the blue arrows press toward 1 from the left and the right, never needing to land on it.



Inputs approach x=1 from both sides (blue arrows). The hollow red circle marks that x never actually reaches 1 – yet the outputs still settle on 2.

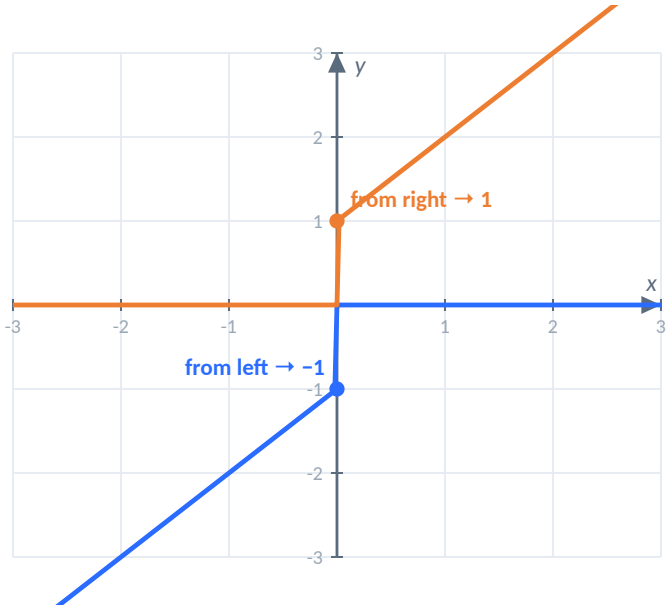
From the left, from the right

Sometimes a function approaches *different* values depending on which side you come from. We split the idea into two **one-sided limits**:

- $\lim_{x \to a^-} f(x)$ – the value approached from the **left** (inputs just below a).
- $\lim_{x \to a^+} f(x)$ – the value approached from the **right** (inputs just above a).

The two-sided limit $\lim_{x \to a} f(x)$ exists **only when both sides agree**. If the left side heads for one number and the right side heads for another, the curve "jumps" and there is no single limit.

One-sided limit



A jump at x=0: the left piece heads for −1, the right piece heads for 1. The sides disagree, so the two-sided limit does not exist.

Continuity, in plain words

A function is **continuous** at a point when nothing surprising happens there: you could draw it without lifting your pen. Formally, three things must line up – the limit exists, the function is actually defined there, and the two match:

$$\lim_{x \to a} f(x) = f(a).$$

Our hole example fails this: the limit is 2 but $f(1)$ does not exist, so there is a tiny break. The jump example fails too. Most everyday functions – lines, parabolas, $\sin x$ – are continuous everywhere, which is why we can usually find a limit just by substituting the value in.

Continuous

Worked examples

Example 1

Find $\lim_{x \to 3} (2x + 1)$.

SOLUTION

1. This is a straight line – smooth and continuous everywhere, with no holes.
2. For a continuous function the limit is just the value: substitute $x = 3$. *no hole → plug in*
3. $2(\,3\,) + 1 = 7$.

The limit is 7. The curve heads straight for 7 and lands on it.

Example 2

Find $\lim_{x \to 1} \dfrac{x^2 - 1}{x - 1}$.

SOLUTION

1. Substituting gives $\frac{0}{0}$, so the formula has a hole here – we cannot just plug in. *0/0 → look closer*
2. Factor the top: $x^2 - 1 = (x - 1)(x + 1)$.
3. For every $x \neq 1$ the $(x - 1)$ cancels, leaving $x + 1$. *cancel the zero*
4. Now the expression is continuous, so substitute: $1 + 1 = 2$.

The limit is 2, exactly the height of the red hole in the figure above.

Example 3

Find $\lim_{x \to 2} \dfrac{x^2 - 4}{x - 2}$.

SOLUTION

1. Substituting gives $\frac{0}{0}$ again – another removable hole.
2. Factor the top as a difference of squares: $x^2 - 4 = (x - 2)(x + 2)$.
3. Cancel the common $(x - 2)$ to leave $x + 2$. *difference of squares*
4. Substitute $x = 2$: $2 + 2 = 4$.

The limit is 4.

Example 4

Find $\lim_{x \to 0} \dfrac{3x}{x}$.

SOLUTION

1. At $x = 0$ this is $\frac{0}{0}$, undefined.
2. But for any $x \neq 0$ the x cancels, leaving the constant 3. *x cancels*
3. A constant function approaches itself, so the limit is 3.

The limit is 3 – the graph is the flat line $y = 3$ with one point missing at $x = 0$.

Example 5

For the jump function $g(x) = x - 1$ when $x < 0$ and $g(x) = x + 1$ when $x > 0$, find the one-sided limits at 0 and decide whether $\lim_{x \to 0} g(x)$ exists.

SOLUTION

1. From the left, use the $x - 1$ piece: as $x \to 0^-$, $g(x) \to 0 - 1 = -1$. *left side*
2. From the right, use the $x + 1$ piece: as $x \to 0^+$, $g(x) \to 0 + 1 = 1$. *right side*
3. The left limit is -1 and the right limit is 1 – they disagree.

Because the two sides differ, the two-sided limit does not exist. The curve jumps.

Example 6

Is $f(x) = x^2$ continuous at $x = 2$?

SOLUTION

1. Find the limit: x^2 is smooth, so $\lim_{x \to 2} x^2 = 2^2 = 4$.
2. Find the value: $f(2) = 2^2 = 4$.
3. The limit and the value are equal, $4 = 4$. *limit = value*

Yes – the limit matches the value, so f is continuous there (no break).

Example 7

Find $\lim_{x \to 3} \dfrac{x^2 - 2x - 3}{x - 3}$.

SOLUTION

1. Substituting $x = 3$ gives $\frac{0}{0}$, so there is a removable hole – factor first. *0/0 → factor*
2. Factor the top into two brackets: $x^2 - 2x - 3 = (x - 3)(x + 1)$.
3. The $(x - 3)$ cancels for every $x \neq 3$, leaving $x + 1$. *cancel the zero*
4. Now substitute $x = 3$ into the simplified form: $3 + 1 = 4$.

The limit is 4.

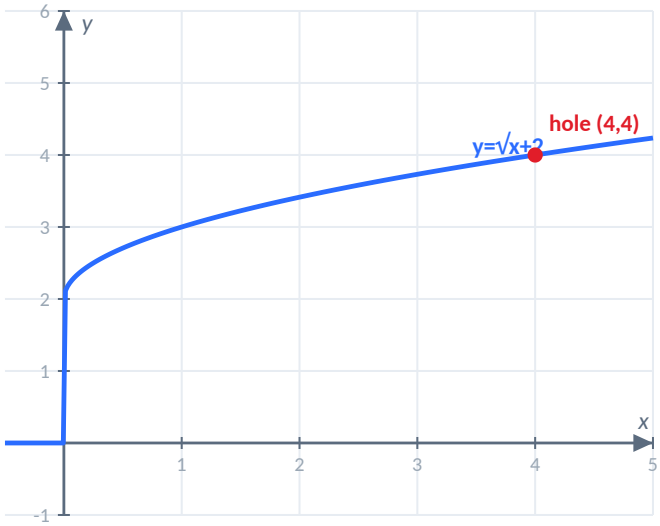
Example 8

Find $\lim_{x\to 4} \dfrac{x-4}{\sqrt{x}-2}$.

SOLUTION

1. Substituting gives $\frac{0}{0}$ again – but the trouble is a square root, so multiply top and bottom by the **conjugate** $\sqrt{x}+2$. ×conjugate
2. The bottom becomes $(\sqrt{x}-2)(\sqrt{x}+2) = x-4$, matching the top.
3. So the fraction is $\dfrac{(x-4)(\sqrt{x}+2)}{x-4}$; cancel the $x-4$ to leave $\sqrt{x}+2$. cancel x−4
4. Substitute $x=4$: $\sqrt{4}+2 = 2+2 = 4$.

The limit is 4 – the conjugate trick cleared the root.



After cancelling, the function equals √x+2 with one point punched out at x=4. The curve still aims straight at the red hole, so the limit is 4.

Example 9

A function is defined in two pieces: $f(x) = x+2$ when $x \le 2$, and $f(x) = 3x-2$ when $x > 2$. Find both one-sided limits at $x=2$, then state $\lim_{x\to 2} f(x)$.

SOLUTION

1. From the left ($x \le 2$) use the $x+2$ piece: as $x \to 2^-$, $f(x) \to 2+2 = 4$. left side
2. From the right ($x > 2$) use the $3x-2$ piece: as $x \to 2^+$, $f(x) \to 3(2)-2 = 4$. right side
3. Both one-sided limits equal 4 – the sides agree, so the two-sided limit exists.

Left $= 4$, right $= 4$, so $\lim_{x\to 2} f(x) = 4$ – the two pieces meet smoothly, no jump.

Note: A limit of $\frac{0}{0}$ is not an answer – it is a signal that says "look closer, probably a hole". Factor and cancel, then try again. The real limit is usually a perfectly ordinary number.

⚡ Mental-Math Shortcut

When you meet a fraction and substitution gives a clean number, **that number is the limit** – done. Only when you get $\frac{0}{0}$ do you need to work: **factor the top and bottom, cancel the part that caused the zero, then substitute again.** For the classic $\frac{x^2 - a^2}{x - a}$, the cancel always leaves $x + a$, so the limit is just $2a$. Spotting "difference of squares on top, matching bracket on the bottom" lets you read the answer off in one step.

🧠 The Logic of It

Why can we be *certain* the limit of $\frac{x^2 - 1}{x - 1}$ is exactly 2, and not 1.9999 or "about 2"? The honest definition is a promise about closeness, phrased as **"as close as you like."**

Claim: for every tolerance you name – say you demand the output be within 0.001 of 2 – I can name a small region around $x = 1$ inside which the function stays that close. Here is the reasoning. For every $x \neq 1$ the function *equals* $x + 1$, so the gap from 2 is

$$|f(x) - 2| = |(x + 1) - 2| = |x - 1|.$$

So the output is within 0.001 of 2 **exactly when** x is within 0.001 of 1. Whatever tolerance you pick, I just match it: stay within that distance of 1 and the promise holds. No tolerance can be too strict.

That is the entire logic of a limit: a *for every* challenge ("get this close") answered by a *there exists* response ("here is the region that does it") – the same universal-then-existential pattern from Chapter 22's quantifiers. Because the challenge can never be beaten, the value the function approaches is pinned to a single number, and that number **is** the limit: exactly 2.

⚠ Common Mistakes

- Thinking the limit *is* the value at the point. They often agree, but a removable hole has a limit with no value – that gap is the whole idea.
- Treating $\frac{0}{0}$ as 0, or as 1, or as "undefined, no answer". It means "factor and cancel first" – the real limit is usually an ordinary number.
- Declaring a limit exists when the two sides disagree. If left heads for one value and right for another, there is no two-sided limit.
- Cancelling $(x - 1)$ and then forgetting the hole is still there. The *function* is undefined at $x = 1$; only the *limit* equals 2.

Quiz

54.1.Q1. In plain words, what does $\lim_{x \to a} f(x) = L$ mean?

54.1.Q2. Find $\lim_{x \to 4} (x + 2)$.

A. 6

B. 4

C. 2

D. undefined

54.1.Q3. Find $\lim_{x \to 1} \dfrac{x^2 - 1}{x - 1}$.

A. 0

B. 1

C. 2

D. undefined

54.1.Q4. Find $\lim_{x \to 3} \dfrac{x^2 - 9}{x - 3}$.

54.1.Q5. A function heads for 5 from the left of $x = 2$ but for 8 from the right. Does $\lim_{x \to 2} f(x)$ exist?

A. Yes, it is 5

B. Yes, it is 8

C. No

D. Yes, it is 6.5

54.1.Q6. What does getting $\frac{0}{0}$ when you substitute tell you to do?

54.1.Q7. True or False: a function must be defined at $x = a$ for $\lim_{x \to a} f(x)$ to exist.

A. True

B. False

54.1.Q8. List the three conditions for f to be continuous at $x = a$.

🐍 From Concept to Code (Python)

A computer cannot reach $x = 1$ in the hole example (it would divide by zero), but it can *sneak up* on it. Evaluate the function at inputs that hug 1 from both sides and watch the outputs settle. This is the table from the lesson, made by a loop.

```
def f(x):
    return (x**2 - 1) / (x - 1)        # undefined exactly at x = 1

for x in [0.9, 0.99, 0.999, 1.001, 1.01, 1.1]:
    print(f"x = {x:<6}  f(x) = {f(x):.4f}")
```

```
output
x = 0.9     f(x) = 1.9000
x = 0.99    f(x) = 1.9900
x = 0.999   f(x) = 1.9990
x = 1.001   f(x) = 2.0010
x = 1.01    f(x) = 2.0100
x = 1.1     f(x) = 2.1000
```

Try it:

1. The outputs are closing in on a single number. Read it off – what is the limit?
2. Add $x = 0.9999$ and $x = 1.0001$ to the list and confirm the outputs get even closer to that number.
3. Change `f` to `(x**2 - 4)/(x - 2)` and loop over values hugging 2 instead. The limit should settle on 4.

Answer Key – Chapter 54 – Limits

54.1.Q1. As x gets closer and closer to a, the value $f(x)$ settles toward L – regardless of what happens exactly at a.

54.1.Q2. 6 – the line is continuous, so just substitute $x = 4$: $4 + 2 = 6$.

54.1.Q3. 2 – factor to $(x - 1)(x + 1)$, cancel, get $x + 1 \to 2$.

54.1.Q4. 6 – $x^2 - 9 = (x - 3)(x + 3)$, cancel to $x + 3$, substitute 3 to get 6.

54.1.Q5. No – the one-sided limits disagree, so there is no two-sided limit (the curve jumps).

54.1.Q6. Look closer: factor the top and bottom, cancel the common part, then substitute again. It usually signals a removable hole.

54.1.Q7. False – the limit can exist (e.g. $= 2$) even where the function has a hole and no value.

54.1.Q8. The limit $\lim_{x \to a} f(x)$ exists, the value $f(a)$ exists, and the two are equal: $\lim_{x \to a} f(x) = f(a)$.

Chapter 55 · Derivatives

How to measure the exact steepness of a curve at a single point – the rate at which something is changing right now – and the simple rule that finds it.

The Idea Behind It

The slope of $y = x^2$ at any point is exactly $2x$: $\frac{d}{dx}\, x^2 = 2x$. *but why?*

The slope at a single point is the slope of a line through *two* points, slid together until they almost touch. So measure the average slope between x and a point a tiny step h away, simplify, then let h shrink to nothing. The rule we lean on is the plain definition of slope, **rise over run**:

$$\boxed{\text{slope} = \frac{f(x+h) - f(x)}{h}}$$

For $f(x) = x^2$ the top needs the square of $x + h$. Expanding it carefully, $(x+h)^2 = x^2 + 2xh + h^2$, so the x^2 at the start cancels the $-x^2$:

$$\frac{(x+h)^2 - x^2}{h} = \frac{x^2 + 2xh + h^2 - x^2}{h} = \frac{2xh + h^2}{h}$$

Every term on top now carries a factor of h, so it cancels the h underneath (this is legal because $h \neq 0$ while the points are still apart):

$$\frac{2xh + h^2}{h} = \frac{h\,(2x + h)}{h} = 2x + h$$

Finally slide the second point home by letting $h \to 0$. The leftover h vanishes, leaving the steepness at that single instant:

$$\therefore \quad \frac{d}{dx}\, x^2 = \lim_{h \to 0} (2x + h) = 2x$$

The same "expand, cancel the h, let $h \to 0$" recipe on x^n leaves $n\, x^{n-1}$ every time, which is the **power rule**.

bring the power down, drop it by one

Everything that moves, grows or cools is *changing*. A car speeds up, a population grows, a coffee cools. **Calculus** is the mathematics of change, and its first big idea is the **derivative**: a way to measure exactly how fast something is changing at one precise instant.

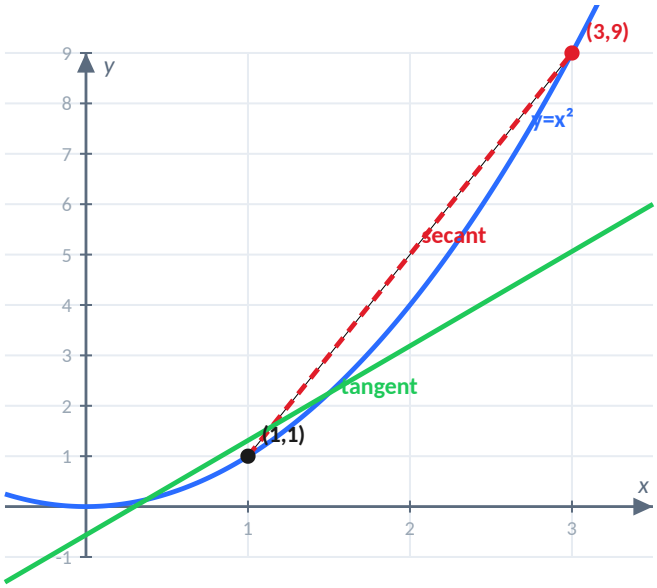
From average speed to a single instant

Suppose a ball's height (in metres) after t seconds is $h = t^2$. Between $t = 1$ and $t = 3$ its **average** rate of change is the familiar "rise over run":

$$\text{average rate} = \frac{\text{change in height}}{\text{change in time}} = \frac{3^2 - 1^2}{3 - 1} = \frac{8}{2} = 4.$$

On a graph, that average rate is the slope of the straight line joining the two points – a **secant** line. But what is the speed at the *single instant* $t = 1$? We cannot divide by zero time. The trick: slide the second point closer and closer

to the first. The secant lines swing towards one special line that just grazes the curve – the **tangent**. Its slope is the instantaneous rate.



The red secant (slope 4) swings towards the green tangent at (1,1) (slope 2) as the points close in.

DERIVATIVE

The **derivative** of a function f, written $f'(x)$ or $\frac{dy}{dx}$, is the slope of the tangent line – the instantaneous rate of change of f at the point x.

The power rule – the everyday shortcut

Finding tangents from scratch is slow, so we use rules. The most useful by far is the **power rule**: to differentiate a power of x, *bring the exponent down to the front, then reduce the exponent by one.*

$$\frac{d}{dx}\, x^n = n\, x^{\,n-1}$$

Two more facts make it complete: the derivative of a **constant** is 0 (a flat line has no slope), and a number multiplying a term just rides along.

Worked examples

Example 1

Differentiate $y = x^2$, then find the slope at $x = 1$.

SOLUTION

1. Power rule: bring the 2 down, reduce the power by one: $\frac{dy}{dx} = 2\, x^{2-1} = 2x$. *power rule*
2. At $x = 1$: slope $= 2(1) = 2$.

$y' = 2x$, **and the slope at** $x = 1$ **is** 2 **– exactly the green tangent above.**

Example 2

Differentiate $y = x^3$.

SOLUTION

1. Bring down the 3, reduce the power: $\frac{dy}{dx} = 3\,x^{3-1} = 3x^2$. down and down

$y' = 3x^2$.

Example 3

Differentiate $y = 5x^2 + 4x - 7$.

SOLUTION

1. Differentiate each term separately. $5x^2 \to 5 \cdot 2x = 10x$.
2. $4x = 4x^1 \to 4 \cdot 1x^0 = 4$.
3. The constant $-7 \to 0$. constant → 0
4. Add the pieces.

$y' = 10x + 4$.

Example 4

A ball's height is $h = t^2$. Find its exact speed at $t = 1$ and $t = 3$ seconds.

SOLUTION

1. Speed is the rate of change of height, so differentiate: $h'(t) = 2t$. speed = h'
2. At $t = 1$: $h'(1) = 2(\,1\,) = 2$ m/s.
3. At $t = 3$: $h'(3) = 2(\,3\,) = 6$ m/s.

The ball is speeding up: 2 m/s at one second, 6 m/s at three seconds.

Example 5

Differentiate $y = 2x^4 - x^2 + 6$.

SOLUTION

1. Term by term. $2x^4 \to 2 \cdot 4x^3 = 8x^3$. bring the 4 down
2. $-x^2 \to -2x = -2x$.
3. The constant $6 \to 0$.

$y' = 8x^3 - 2x$.

Example 6

For $y = x^3$, find the slope of the tangent at $x = 2$.

SOLUTION

1. First differentiate: $x^3 \to 3x^2$. power rule
2. Now substitute $x = 2$: slope $= 3(2)^2 = 3 \cdot 4 = 12$.

$y' = 3x^2$, **so the slope at** $x = 2$ **is** 12.

Example 7

Differentiate $y = 6x^5 - 3x^2 + 2x - 9$.

SOLUTION

1. Differentiate each term on its own, then add the results.
2. $6x^5 \to 6 \cdot 5x^4 = 30x^4$. bring the 5 down
3. $-3x^2 \to -3 \cdot 2x = -6x$.
4. $2x \to 2$ (since $x^1 \to 1$), and the constant $-9 \to 0$. constant → 0

$y' = 30x^4 - 6x + 2$.

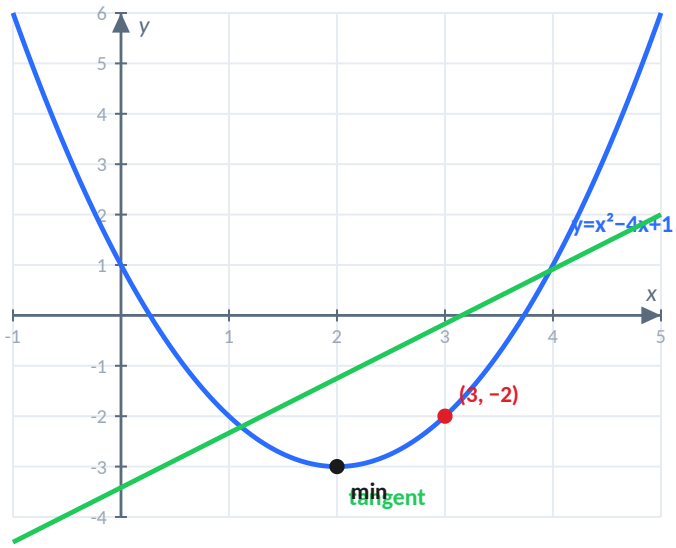
Example 8

Find the slope of $y = x^2 - 4x + 1$ at $x = 3$, and the point the tangent touches.

SOLUTION

1. Differentiate term by term: $x^2 \to 2x$, $-4x \to -4$, constant $\to 0$, so $y' = 2x - 4$.
   power rule, term by term
2. Slope at $x = 3$: $y'(3) = 2(3) - 4 = 6 - 4 = 2$.
3. The touch-point needs the y-value too: $y(3) = 3^2 - 4(3) + 1 = 9 - 12 + 1 = -2$.

$y' = 2x - 4$**; the slope at** $x = 3$ **is** 2**, where the tangent grazes the point** $(3, -2)$**.**



The green tangent to y=x²−4x+1 at (3,−2) has slope 2 (the derivative 2x−4 at x=3). The curve bottoms out where 2x−4=0, at x=2.

Example 9

For $y = x^2$, at what value of x does the curve have slope 10?

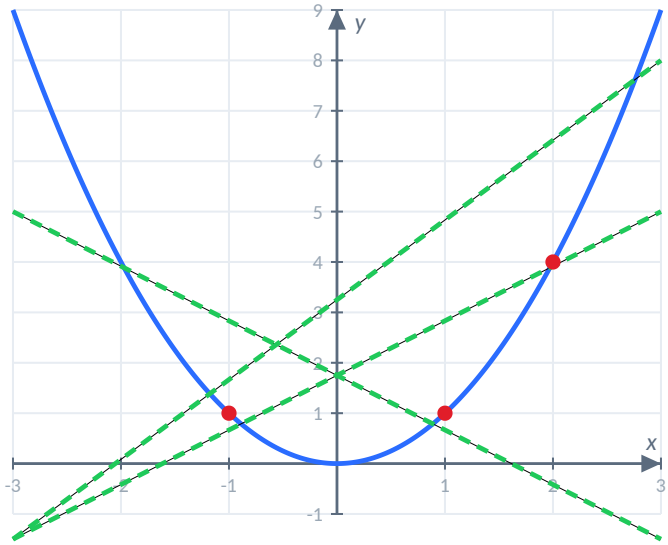
SOLUTION

1. The slope at any point is the derivative: $x^2 \to y' = 2x$. power rule
2. Set that slope equal to the wanted value: $2x = 10$. slope = 10
3. Solve for x: divide both sides by 2, giving $x = 5$.

The slope is 10 exactly at $x = 5$ (there $2x = 2 \cdot 5 = 10$).

Note: Read $\frac{dy}{dx}$ as "the rate of change of y with respect to x". It is a single symbol for a slope, not a fraction you can split apart.

The derivative is itself a function: feed it any x and it returns the slope there. Below, the tangent has been drawn at three points of $y = x^2$ – notice the slope $(2x)$ grows as we move right.



Tangents to y=x² at x=−1, 1, 2 have slopes −2, 2, 4 – always 2x.

⚡ Mental-Math Shortcut

The power rule in your head: **"down and down."** The exponent comes *down* to the front as a multiplier, and the exponent itself goes *down* by one. So $x^4 \to 4x^3$, $x^7 \to 7x^6$, and a lone $x = x^1 \to 1x^0 = 1$. Constants vanish.

Why is the instantaneous rate a *single, definite* number and not just a vague idea? This is a **limit** argument, and it follows a tidy logical pattern. Compute the average slope of $y = x^2$ between $x = 1$ and a nearby point $x = 1 + s$:

$$\text{slope} = \frac{(1+s)^2 - 1^2}{s} = \frac{1 + 2s + s^2 - 1}{s} = \frac{2s + s^2}{s} = 2 + s.$$

Now reason it through: *for every* tiny step s (no matter how small), the slope equals $2 + s$. As s shrinks toward 0, the slope is squeezed toward 2 and can be made as close to 2 as we like. There is no other value it could approach. Therefore the slope at $x = 1$ **must** be exactly 2. That "for every… therefore" structure – a universal claim pinned down by squeezing – is the same logical move (Chapter 22's *for all*) that underpins all of calculus.

⚠ Common Mistakes

- Forgetting to reduce the exponent: x^3 differentiates to $3x^2$, **not** $3x^3$.
- Thinking the derivative of a constant is the constant. A flat line has slope 0, so $\frac{d}{dx}(7) = 0$.
- Differentiating x (which is x^1) to 0. It is 1, because $1 \cdot x^0 = 1$.
- Reading $\frac{dy}{dx}$ as a fraction and "cancelling" the d's. It is one symbol meaning "slope".

Quiz

55.1.Q1. Differentiate $y = x^5$.

A. $5x^4$

B. $5x^6$

C. x^4

D. $4x^5$

55.1.Q2. What is the derivative of a constant, such as $y = 12$?

55.1.Q3. Differentiate $y = 3x^2$.

55.1.Q4. Find $\frac{dy}{dx}$ for $y = x^2 + x$.

55.1.Q5. For $y = x^2$, what is the slope of the tangent at $x = 4$?

A. 16

B. 8

C. 4

D. 2

55.1.Q6. A car's position is $s = t^2$. What is its speed $s'(t)$?

55.1.Q7. True or False: the derivative of $y = x^2$ is the same number everywhere.

A. True

B. False

55.1.Q8. Differentiate $y = 4x^3 - 2x + 9$.

🐍 From Concept to Code (Python)

A computer can *estimate* a derivative without any algebra: just measure the average slope over a tiny step h and let h be small. This is the secant-becomes-tangent idea in code.

```
def slope(f, x, h=1e-6):
    return (f(x + h) - f(x)) / h     # rise over a tiny run

f = lambda x: x**2                   # the curve y = x²
print(round(slope(f, 1), 4))         # tangent slope at x = 1
print(round(slope(f, 3), 4))         # ...and at x = 3
```

```
output
2.0
6.0
```

Try it:

1. The exact derivative of x^2 is $2x$. Check the code gives $\approx 2x$ at $x = 5$ and $x = 10$.
2. Change `f` to `lambda x: x**3` and confirm the slope at $x = 2$ is close to 12 (since $3x^2 = 12$).

Answer Key – Chapter 55 – Derivatives

55.1.Q1. $5x^4$ – bring the 5 down, reduce the power by one.

55.1.Q2. 0 – a constant never changes, so its rate of change is zero.

55.1.Q3. $y' = 6x$ – the 3 rides along while $x^2 \to 2x$, giving $3 \cdot 2x$.

55.1.Q4. $2x + 1$ – differentiate each term: $x^2 \to 2x$ and $x \to 1$.

55.1.Q5. 8 – the derivative is $2x$, and $2(4) = 8$.

55.1.Q6. $2t$ – speed is the derivative of position.

55.1.Q7. False – it is $2x$, which changes with x (steeper as x grows).

55.1.Q8. $12x^2 - 2$ – term by term: $4x^3 \to 12x^2$, $-2x \to -2$, $9 \to 0$.

Chapter 56 · Rates of Change & Motion

*How the derivative stops being abstract algebra and becomes a real-world **rate** – the speed of a car, the spread of a rumour, the cooling of coffee – and the difference between an average rate over an interval and the instantaneous rate at a single moment.*

The Idea Behind It

The speed shown on a speedometer at one instant is the derivative of position: instantaneous velocity $= s'(t)$. *but why?*

Speed looks like it needs two clock readings: distance covered divided by time taken. At a single instant no time passes, so that division seems to collapse into the nonsense $\frac{0}{0}$. The escape is to measure the speed over a short window of width h, then shrink the window toward nothing. We lean on the everyday meaning of speed, **distance over time**:

$$\boxed{\text{average velocity over } [t,\, t+h] = \frac{s(t+h) - s(t)}{h}}$$

Take a concrete trip, $s(t) = t^2$, and watch the window from t. Expanding $(t+h)^2 = t^2 + 2th + h^2$, the t^2 cancels and the surviving terms all share a factor of h:

$$\frac{(t+h)^2 - t^2}{h} = \frac{t^2 + 2th + h^2 - t^2}{h} = \frac{2th + h^2}{h} = 2t + h$$

That holds for *every* window width $h \neq 0$. Now close the window by letting $h \to 0$: the average rate is squeezed onto a single number, the rate at that one instant.

$$\therefore\ v(t) = \lim_{h \to 0} \frac{s(t+h) - s(t)}{h} = \lim_{h \to 0} (2t + h) = 2t = s'(t)$$

So instantaneous velocity is not a division by zero, it is the limit of honest average velocities, which is exactly the derivative of position. *average rate, with the window shrunk to an instant*

A derivative is a slope, but slope is just the abstract shadow of something you feel every day: a **rate of change**. When a speedometer reads 60, when a savings account grows, when a balloon deflates, a derivative is quietly at work. This chapter reads the slope *in context* and gives it units and meaning.

Position, then speed

The cleanest example is motion. Suppose a car's distance from home (in metres) after t seconds is a function $s(t)$. The car's **speed** is how fast that distance is changing – which is exactly the derivative $s'(t)$. Distance is the "what", speed is the "how fast the what is changing".

$$\text{speed} = \frac{ds}{dt} = s'(t).$$

The same pattern repeats across every subject. The rate of change of a *population* is its growth rate; of *temperature*, the rate of heating or cooling; of *money*, the income or burn rate. One idea, many costumes.

RATE OF CHANGE

A **rate of change** is the derivative of a quantity with respect to time (or another variable): how many units of output you gain per one unit of input. Its sign tells direction (rising or falling) and its size tells how fast.

Rate of change Velocity

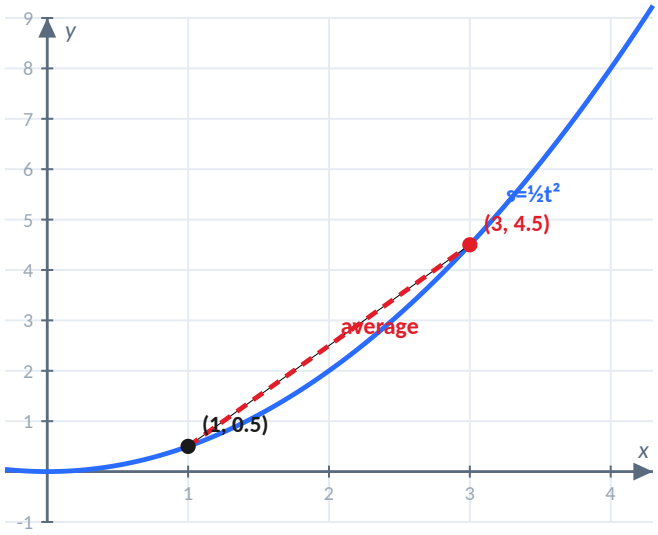
Average rate: rise over run

Over a stretch of time, the **average rate** is the total change divided by the time taken – the plain "rise over run" between two points on the graph. If the car is at $s(t_1)$ and later at $s(t_2)$:

$$\text{average rate} = \frac{s(t_2) - s(t_1)}{t_2 - t_1}.$$

On the graph this is the slope of the straight **secant** line joining the two readings. It is an honest summary of the whole interval, but it hides the detail – the car may have sped up and slowed down in between.

Average rate

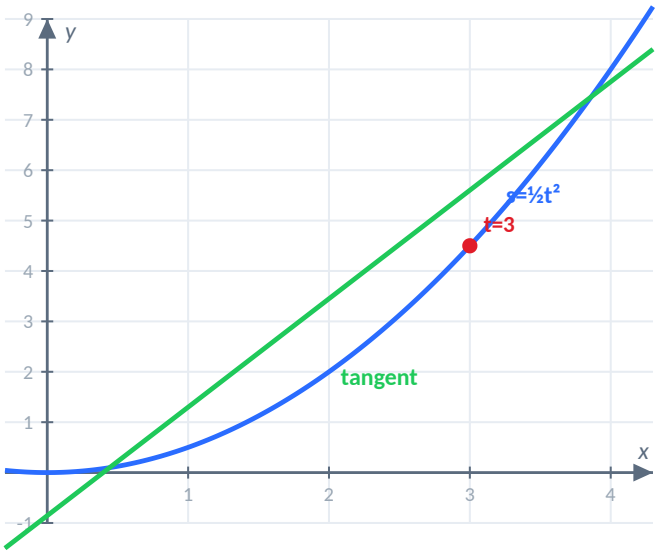


Position s=½t². The red secant from t=1 to t=3 has slope (4.5−0.5)/(3−1)=2, the average speed over those two seconds.

Instantaneous rate: the speedometer

The average rate cannot tell you the speed at *one* instant – a single point has no interval to divide over. So we do the same trick as Chapter 55: slide the second reading toward the first. The secant lines swing toward the **tangent**, and its slope is the **instantaneous rate** – the number a speedometer shows at that exact moment. That tangent slope is precisely the derivative.

Instantaneous rate



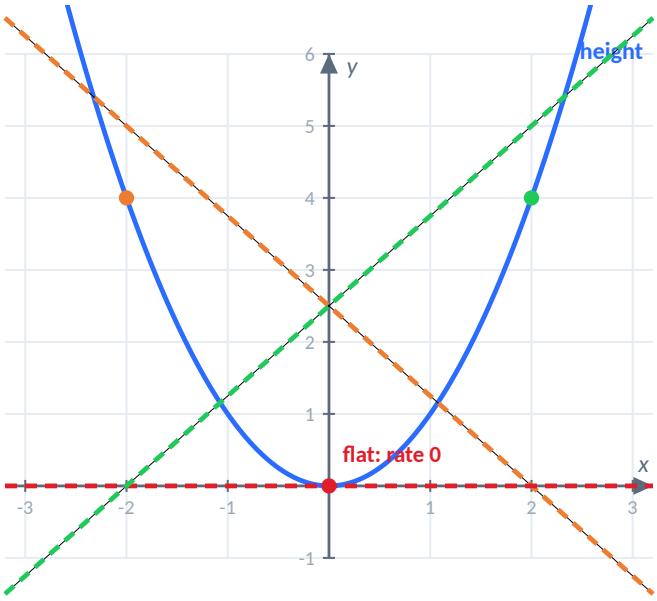
The green tangent just grazes the curve at t=3. Its slope is s′(3)=3, the instantaneous speed (3 m/s) at that exact second.

For $s = \frac{1}{2}t^2$ the power rule gives $s'(t) = t$. So at $t = 3$ the instantaneous speed is $s'(3) = 3$ m/s – steeper than the average of 2 m/s over $[1, 3]$, because the car was still speeding up. Average smooths; instantaneous pinpoints.

Reading the sign and size of a slope

Because a rate is a slope, its *shape* tells a story at a glance:

- A **positive** slope means the quantity is rising (car moving away, population growing).
- A **negative** slope means it is falling (returning home, coffee cooling, account draining).
- A **zero** slope means it is momentarily unchanging (paused at a traffic light, a population at its peak).
- A **steeper** slope – up or down – means a faster rate; a gentle slope means a slow one.



On h=t²: falling-then-rising. The orange tangent (slope −4) shows a fast decrease, the red one (slope 0) a momentary standstill, the green one (slope 4) a fast increase.

Worked examples

Example 1

A car's position is $s(t) = t^2$ metres. Find its average speed between $t = 2$ and $t = 5$ seconds.

SOLUTION

1. Find the two positions: $s(2) = 2^2 = 4$ m and $s(5) = 5^2 = 25$ m.
2. Average rate is rise over run: $\dfrac{25 - 4}{5 - 2} = \dfrac{21}{3}$. *rise over run*
3. Simplify: $\dfrac{21}{3} = 7$.

The average speed is 7 m/s over those three seconds.

Example 2

For the same car $s(t) = t^2$, find the instantaneous speed at $t = 5$.

SOLUTION

1. Instantaneous speed is the derivative: differentiate $s(t) = t^2$ to get $s'(t) = 2t$. *power rule*
2. Evaluate at the instant: $s'(5) = 2(5)$. *plug in the moment*
3. So $s'(5) = 10$.

The instantaneous speed at $t = 5$ is 10 m/s – faster than the 7 m/s average, because the car kept accelerating.

Example 3

A balloon's volume is $V(t) = 30 - 2t$ litres as it deflates. Find the rate of change of volume.

SOLUTION

1. Differentiate term by term: the constant $30 \to 0$, and $-2t \to -2$. *constant vanishes*
2. So $V'(t) = -2$.

The volume changes at -2 litres per second. The negative sign means it is shrinking, and the size 2 means 2 litres leave each second.

Example 4

A thrown ball has height $h(t) = 20t - 5t^2$ metres. Find its velocity at $t = 1$ s and at $t = 3$ s.

SOLUTION

1. Differentiate: $20t \to 20$ and $-5t^2 \to -10t$, so $h'(t) = 20 - 10t$. power rule
2. At $t = 1$: $h'(1) = 20 - 10(\,1\,) = 10$ m/s (positive, so rising).
3. At $t = 3$: $h'(3) = 20 - 10(\,3\,) = -10$ m/s (negative, so falling back down).

The ball rises at 10 m/s after one second, then falls at 10 m/s after three seconds.

Example 5

For that ball $h(t) = 20t - 5t^2$, at what moment is it momentarily at rest (velocity zero), at the top of its flight?

SOLUTION

1. Velocity is $h'(t) = 20 - 10t$. At rest means $20 - 10t = 0$. set velocity = 0
2. Solve $20 - 10t = 0$, so $10t = 20$, giving $t = 2$.

At $t = 2$ s the velocity is 0 – the ball is momentarily still at the peak before falling.

Example 6

A population grows as $P(t) = 100 + 5t^2$ (people, t in years). Find the average growth rate over the first 4 years and the instantaneous rate at $t = 4$.

SOLUTION

1. Average: $P(0) = 100$ and $P(4) = 100 + 5(16) = 180$, so $\dfrac{180 - 100}{4 - 0} = \dfrac{80}{4} = 20$ people/year.

   rise over run
2. Instantaneous: differentiate $P(t) = 100 + 5t^2$ to get $P'(t) = 10t$. power rule
3. Evaluate: $P'(4) = 10(\,4\,) = 40$ people/year.

Average growth is 20 people/year, but the instantaneous rate at year 4 is 40 people/year – growth is accelerating.

Example 7

A drone's height is $h(t) = t^3 - 6t^2 + 9t$ metres (t in seconds). Find its velocity, then its velocity at $t = 1$ s.

SOLUTION

1. Velocity is the derivative of height. Differentiate term by term: $t^3 \to 3t^2$, $-6t^2 \to -12t$, $9t \to 9$.
   power rule each term
2. So $h'(t) = 3t^2 - 12t + 9$.
3. At $t = 1$: $h'(1) = 3(1)^2 - 12(1) + 9 = 3 - 12 + 9 = 0$. *plug in t=1*

The velocity is $h'(t) = 3t^2 - 12t + 9$, and at $t = 1$ s it is 0 m/s – the drone is momentarily hovering.

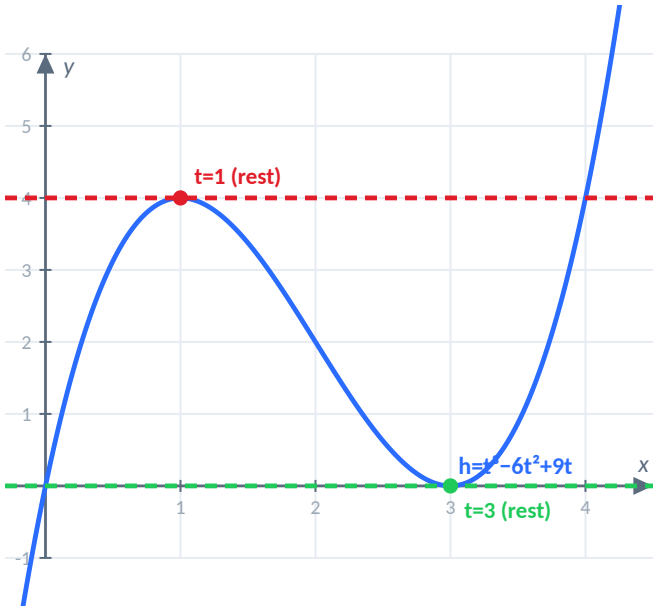
Example 8

For that same drone $h(t) = t^3 - 6t^2 + 9t$, at which times is it momentarily at rest (velocity zero)?

SOLUTION

1. At rest means $h'(t) = 0$. From Example 7, $h'(t) = 3t^2 - 12t + 9$. *set velocity = 0*
2. Factor out the common 3: $3(t^2 - 4t + 3) = 0$, so $t^2 - 4t + 3 = 0$.
3. Factor the quadratic: $(t - 1)(t - 3) = 0$, giving $t = 1$ or $t = 3$. *two rest moments*

The drone is at rest at $t = 1$ s and $t = 3$ s – once on the way up, once at the top of a dip.



The drone's height curve is flat (zero velocity) at t=1 s (a local peak) and t=3 s (a local trough) – exactly where h′(t)=0.

Example 9

Coffee cools so its temperature is $T(t) = 90 - 4t + 0.1t^2$ degrees (t in minutes). Find the rate of cooling at $t = 5$ min, and say whether it is cooling faster or slower than at $t = 10$ min.

SOLUTION

1. Differentiate term by term: constant $90 \to 0$, $-4t \to -4$, $0.1t^2 \to 0.2t$, so $T'(t) = -4 + 0.2t$.
   power rule
2. At $t = 5$: $T'(5) = -4 + 0.2(5) = -4 + 1 = -3$ degrees/min. negative = cooling
3. At $t = 10$: $T'(10) = -4 + 0.2(10) = -4 + 2 = -2$ degrees/min.

At 5 min the coffee cools at -3 degrees/min; at 10 min only -2 degrees/min – cooling slows down as the coffee nears room temperature (the rate climbs toward 0).

Note: Average and instantaneous rates answer different questions. "How fast on the whole trip?" is the average (secant slope). "How fast right now?" is the instantaneous rate (tangent slope, the derivative). They only match when the rate is constant – a straight-line graph.

⚡ Mental-Math Shortcut

To get an **average** rate, you do not need calculus at all – just **(end value − start value) ÷ (time taken)**, the plain rise over run. To get the **instantaneous** rate, **differentiate, then plug in the moment.** A quick gut-check: on a curve that bends upward, the instantaneous rate at the right end is *bigger* than the average across the interval (the curve is steeper there); on a curve bending downward it is *smaller*. If your two numbers come out wildly different in the wrong direction, recheck a sign.

🧠 The Logic of It

Why is it *legitimate* to speak of a speed at a single instant, when speed seems to be defined as distance divided by time – and at one instant no time passes, giving the nonsense $\frac{0}{0}$?

Reason it through. Average speed over a window of length w starting at $t = 2$, for $s(t) = t^2$, is

$$\frac{s(2+w) - s(2)}{w} = \frac{(2+w)^2 - 4}{w} = \frac{4 + 4w + w^2 - 4}{w} = \frac{4w + w^2}{w} = 4 + w.$$

Now the key move: this holds for *every* window width w, however tiny (as long as $w \neq 0$). As w shrinks toward 0, the average speed $4 + w$ is squeezed toward 4 and can be forced as close to 4 as anyone demands – the "as close as you like" promise from Chapter 54. No other value is approached.

So "instantaneous speed" is not a single division by zero; it is the **one value the shrinking averages converge on**. That convergence is guaranteed and unique, so the instant *does* have a definite speed: exactly 4 m/s. The logic is the limit's *for every challenge, there is a matching response* (Chapter 22's quantifiers again) – which is why a speedometer can honestly display one number.

⚠ Common Mistakes

- Confusing average and instantaneous rate. Average uses two points (a secant); instantaneous uses one point and the derivative (a tangent). They agree only for straight-line graphs.
- Dropping the sign of a rate. A velocity of -10 is not the same as 10: the minus means moving the other way (falling, draining, cooling).
- Computing an average rate as $\frac{s(t_2)}{s(t_1)}$ (a ratio) instead of $\frac{s(t_2)-s(t_1)}{t_2-t_1}$ (a difference over a difference).
- Forgetting the units. A rate is always "output units per one input unit" – metres per second, people per year – not a bare number.
- Assuming zero velocity means the object is gone or stopped forever. It can be a momentary pause, like a ball at the very top of its flight.

📝 Quiz

56.1.Q1. What real-world quantity is the derivative of *position*?

A. Distance

B. Velocity (speed)

C. Acceleration

D. Time

56.1.Q2. A car is at 10 m at $t = 1$ s and at 40 m at $t = 4$ s. What is its average speed?

56.1.Q3. For $s(t) = t^2$, what is the instantaneous speed $s'(t)$?

A. t^2

B. $2t$

C. t

D. 2

56.1.Q4. A tank drains so its volume changes at a rate of -3 litres per minute. What does the minus sign mean?

56.1.Q5. On a position graph, what does a flat (zero-slope) point mean?

56.1.Q6. A ball has $h(t) = 20t - 5t^2$. Find its velocity $h'(t)$.

56.1.Q7. Using $h'(t) = 20 - 10t$, at what time is the ball momentarily at rest?

A. $t = 0$

B. $t = 1$

C. $t = 2$

D. $t = 4$

56.1.Q8. True or False: the average and instantaneous rates are always equal.

A. True

B. False

🐍 From Concept to Code (Python)

You can read an *instantaneous* rate straight from real data without any algebra: take two measurements a tiny moment apart and divide the change in value by the change in time. The closer the readings, the better the estimate of the true rate – the secant-becomes-tangent idea, applied to numbers a sensor might record.

```
def rate(s, t, dt=1e-6):
    return (s(t + dt) - s(t)) / dt      # tiny change in value / tiny change in time

s = lambda t: t**2                      # car position: s = t² metres

print("avg over [2,5]:", (s(5) - s(2)) / (5 - 2))   # average speed
print("instant at t=5:", round(rate(s, 5), 4))      # speed right now
```

```
output
avg over [2,5]: 7.0
instant at t=5: 10.0
```

Try it:

1. The exact derivative of t^2 is $2t$. Check the instantaneous estimate at $t = 3$ is close to 6, and at $t = 10$ close to 20.
2. Change `s` to `lambda t: 20*t - 5*t**2` (the thrown ball) and find the instantaneous rate at $t = 2$. It should be about 0 – the top of the flight.
3. Make `dt` larger, say `0.5`, and watch the instantaneous estimate drift away from the exact value. Smaller `dt` means a sharper reading.

Answer Key – Chapter 56 – Rates of Change & Motion

56.1.Q1. Velocity (speed) – it is the rate at which position changes.

56.1.Q2. 10 m/s – rise over run: $(40 - 10)/(4 - 1) = 30/3 = 10$.

56.1.Q3. $2t$ – differentiate the position to get the velocity.

56.1.Q4. The volume is decreasing – the tank is losing 3 litres each minute.

56.1.Q5. The object is momentarily not moving – its instantaneous velocity is zero at that instant.

56.1.Q6. $20 - 10t$ – differentiate: $20t \to 20$ and $-5t^2 \to -10t$.

56.1.Q7. $t = 2$ – set $20 - 10t = 0$ and solve.

56.1.Q8. False – they are equal only when the rate is constant (a straight-line graph). On a curve they generally differ.

Chapter 57 · Optimisation

How to find the biggest or smallest value a quantity can reach – the cheapest cost, the largest area, the highest profit – using one beautifully simple fact: the peaks and valleys of a curve are exactly the places where its slope is zero.

The Idea Behind It

At the very top of a smooth hill (or bottom of a valley) the slope is exactly zero: a maximum or minimum can occur only where $f'(x) = 0$. *but why?*

Think about how the slope behaves as you cross a smooth peak. On the way up the curve is *rising*, so the slope is positive; just past the top, on the way down, the curve is *falling*, so the slope is negative. The fact we lean on is the plain reading of the derivative as the **slope of the tangent**:

$$\boxed{f'(x) > 0 \Rightarrow \text{rising}, \qquad f'(x) < 0 \Rightarrow \text{falling}}$$

So as x passes through the peak, the slope changes from positive to negative :

$$\underbrace{f'(x) > 0}_{\text{just before}} \longrightarrow \underbrace{f'(x) < 0}_{\text{just after}}$$

A *smooth* slope cannot jump straight from positive to negative without passing through every value between, so it must hit 0 at the turning point itself. (Try to keep a positive slope and the curve would still be climbing, meaning you had not reached the top yet, a contradiction.)

$$\therefore \text{ at a smooth peak or valley}, \quad f'(x) = 0$$

This is why optimisation begins by solving $f'(x) = 0$: it finds the flat tangents, the only places a maximum or minimum can hide. (Zero slope is *necessary*, not a guarantee, so you still classify each one.)

flat at the very top

So much of the useful world is about getting the *most* or the *least*: the most area from a fixed fence, the least metal in a can, the highest a ball will fly, the lowest a cost can fall. **Optimisation** is the art of finding those extreme values, and the derivative hands us a single, reliable key.

Peaks and valleys are flat

Walk along a curve. As you climb toward a peak the slope is positive; just past it, on the way down, the slope is negative. Right *at* the top the slope must pass through zero – for an instant the curve is level. The same is true at the bottom of a valley. So the highest and lowest points share one signature: a **flat tangent**, slope zero.

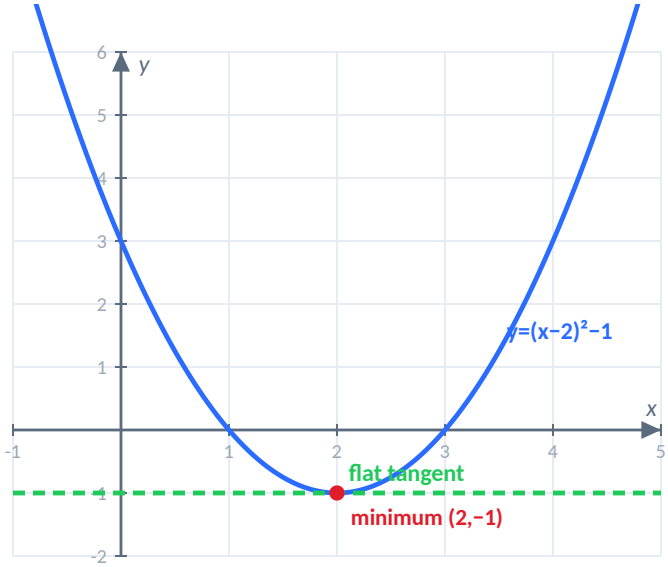
$$\text{a maximum or minimum can occur only where } f'(x) = 0.$$

This gives a recipe. The extreme values hide among the points where the derivative is zero – the **critical points**. Find those points and we have found every candidate.

CRITICAL POINT

A **critical point** of f is an input x where the derivative is zero, $f'(x) = 0$ – a place where the tangent is flat. Maxima and minima live only at critical points.

Critical point Maximum Minimum



A valley. The slope is negative on the left, positive on the right, and exactly zero at the bottom (2,−1), where the green tangent is perfectly flat.

The three-step recipe

To optimise a function, follow the same three steps every time:

1. **Differentiate** to get $f'(x)$, the slope formula.
2. **Set it to zero** and solve $f'(x) = 0$ to find the critical point(s).
3. **Classify** each one as a maximum or a minimum (a peak or a valley).

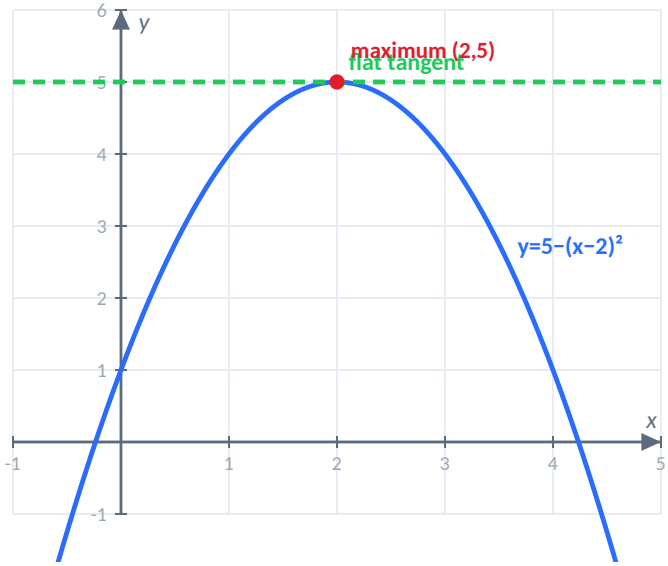
Step 3 matters because slope-zero alone does not say *which* kind of flat spot you have found.

Is it a peak or a valley?

Two easy tests tell a maximum from a minimum:

- **Slope-sign test.** Check the slope just before and just after the critical point. Slope going $+$ then $-$ means a **peak** (rising then falling). Going $-$ then $+$ means a **valley** (falling then rising).
- **Second-derivative test.** Differentiate again to get $f''(x)$. If $f'' > 0$ the curve is cupped upward (a valley, **minimum**); if $f'' < 0$ it is cupped downward like a hill (a **maximum**).

Second derivative



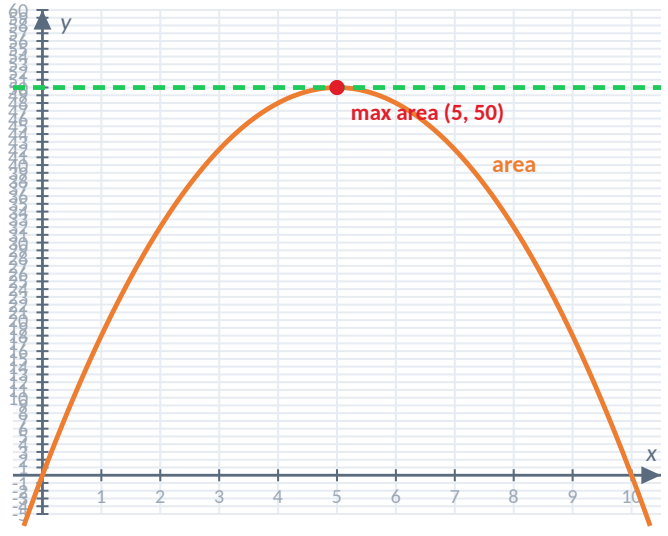
A hill. The slope runs positive then negative, hitting zero at the peak (2,5). The downward cup (second derivative negative) confirms it is a maximum.

A real-world shape

Suppose you have 20 metres of fence to make a rectangular pen against a wall (so the wall is one long side and you fence the other three). If the two short sides each have length x, the long side is $20 - 2x$, and the area is

$$A(x) = x(20 - 2x) = 20x - 2x^2.$$

This is an upside-down parabola: area starts at zero (no width), rises to a single best value, then falls back to zero. The peak is the largest possible pen – and we find it where $A'(x) = 0$.



Pen area A=20x−2x². It peaks at x=5 m, giving the largest area of 50 m². Wider or narrower than 5 m and you enclose less.

Worked examples

Example 1

Find the minimum of $f(x) = x^2 - 4x + 7$.

SOLUTION

1. Differentiate: $f'(x) = 2x - 4$. *power rule*
2. Set it to zero: $2x - 4 = 0$, so $x = 2$ – the critical point. *set f'=0*
3. Classify with the second derivative: $f''(x) = 2 > 0$, an upward cup, so this is a minimum.
4. Find the value there: $f(2) = 2^2 - 4(2) + 7 = 4 - 8 + 7 = 3$.

The minimum is 3, reached at $x = 2$ (the bottom of the valley).

Example 2

Find the maximum of $f(x) = 6x - x^2$.

SOLUTION

1. Differentiate: $f'(x) = 6 - 2x$. *power rule*
2. Set to zero: $6 - 2x = 0$, so $x = 3$. *set f'=0*
3. Classify: $f''(x) = -2 < 0$, a downward cup, so this is a maximum.
4. Value: $f(3) = 6(3) - 3^2 = 18 - 9 = 9$.

The maximum is 9, reached at $x = 3$ (the top of the hill).

Example 3

A ball's height is $h(t) = 20t - 5t^2$ metres. Find the greatest height it reaches.

SOLUTION

1. Differentiate to get the velocity: $h'(t) = 20 - 10t$. *power rule*
2. At the top the ball is momentarily still, so set $20 - 10t = 0$: this gives $t = 2$ s. *set h'=0*
3. The velocity goes $+$ (rising) then $-$ (falling), confirming a peak.
4. Greatest height: $h(2) = 20(2) - 5(2^2) = 40 - 20 = 20$.

The ball reaches a maximum height of 20 metres, at $t = 2$ seconds.

Example 4

You have 20 m of fence for a three-sided rectangular pen against a wall. What dimensions give the largest area?

SOLUTION

1. Let each short side be x; the long side is $20 - 2x$, so $A(x) = 20x - 2x^2$.
2. Differentiate: $A'(x) = 20 - 4x$. power rule
3. Set to zero: $20 - 4x = 0$, so $x = 5$ m. set A'=0
4. Classify: $A''(x) = -4 < 0$, a hill, so $x = 5$ gives a maximum (not a minimum).
5. Long side $= 20 - 2(5) = 10$ m; area $= 5 \times 10 = 50$ m².

The largest pen is 5 m by 10 m, enclosing 50 m² – exactly the peak in the orange figure.

Example 5

Find the critical point of $f(x) = x^2 - 6x + 5$ and state whether it is a max or min.

SOLUTION

1. Differentiate: $f'(x) = 2x - 6$. power rule
2. Set to zero: $2x - 6 = 0$, so $x = 3$. set f'=0
3. Second derivative: $f''(x) = 2 > 0$, an upward cup.

There is a single critical point at $x = 3$, and it is a minimum (the parabola opens upward).

Example 6

Of all rectangles with perimeter 24, which has the greatest area? (Let the width be x.)

SOLUTION

1. Perimeter 24 means width + height = 12, so the height is $12 - x$.
2. Area: $A(x) = x(12 - x) = 12x - x^2$.
3. Differentiate and set to zero: $A'(x) = 12 - 2x = 0$, so $x = 6$. set A'=0
4. Classify: $A''(x) = -2 < 0$, a maximum. Height $= 12 - 6 = 6$.

The best rectangle is 6×6 – a square – with area 36. Among fixed-perimeter rectangles, the square always wins.

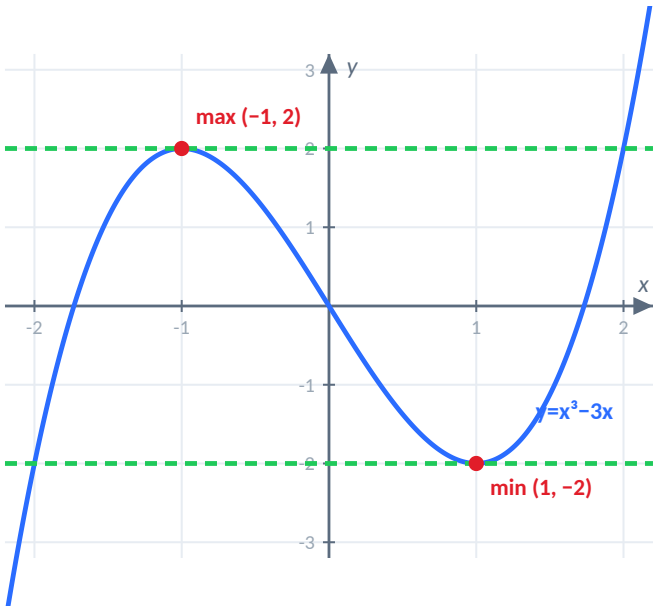
Example 7

A cubic has two flat spots. Find both critical points of $f(x) = x^3 - 3x$ and classify each.

SOLUTION

1. Differentiate: $f'(x) = 3x^2 - 3$. *power rule*
2. Set to zero: $3x^2 - 3 = 0$ gives $x^2 = 1$, so $x = -1$ and $x = 1$ – two critical points this time. *set f'=0*
3. Second derivative: $f''(x) = 6x$. *differentiate again*
4. At $x = -1$: $f'' = 6(-1) = -6 < 0$, a downward cup, so a **maximum**; value $f(-1) = (-1)^3 - 3(-1) = 2$.
5. At $x = 1$: $f'' = 6(1) = 6 > 0$, an upward cup, so a **minimum**; value $f(1) = 1 - 3 = -2$.

A local maximum at $x = -1$ (value 2) and a local minimum at $x = 1$ (value -2) – the same curve carries both.



The cubic x^3-3x from Example 7 has two flat tangents: a hill-top at (−1, 2) and a valley-floor at (1, −2). One sign test on $f''=6x$ tells them apart.

Example 8

A closed box with a square base must hold 27 cm³. Its base is x by x and its height is h. What dimensions use the least material (smallest surface area)?

SOLUTION

1. The fixed volume pins the height to the base: $x^2 h = 27$, so $h = \dfrac{27}{x^2}$. *use the constraint*
2. Surface area of a closed box = two square ends + four sides $= 2x^2 + 4xh$. Substitute h: $S(x) = 2x^2 + 4x \cdot \dfrac{27}{x^2} = \boxed{2x^2 + \dfrac{108}{x}}$. *one variable now*
3. Differentiate, treating $\dfrac{108}{x} = 108x^{-1}$ so it goes to $-108x^{-2}$: $S'(x) = \boxed{4x - \dfrac{108}{x^2}}$.

   power rule, both terms
4. Set to zero: $4x = \dfrac{108}{x^2}$, so $4x^3 = 108$, giving $x^3 = 27$ and $x = \boxed{3}$ cm. *set S'=0*
5. Classify: $S''(x) = 4 + \dfrac{216}{x^3} > 0$ for every $x > 0$, an upward cup, so this critical point is a genuine **minimum**.
6. Height: $h = \dfrac{27}{3^2} = 3$ cm – the least-material box is a perfect cube.

The box is $3 \times 3 \times 3$ cm – a cube – using the least material. As with the square pen, symmetry wins: $x = \boxed{3}$ cm.

Example 9

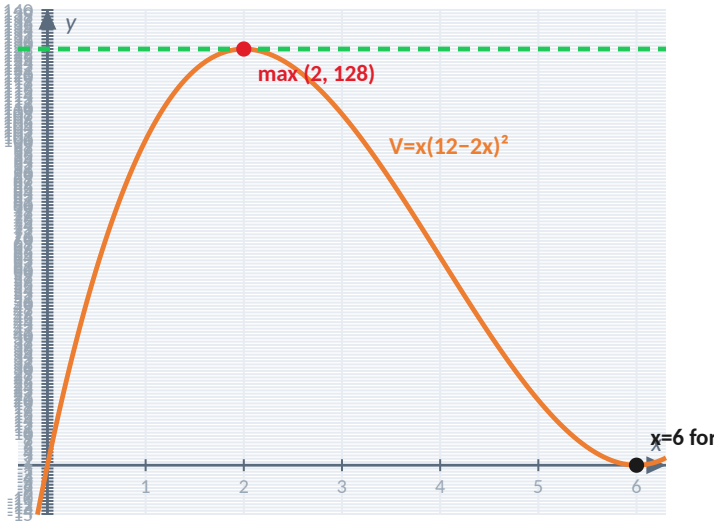
An open-topped box is folded from a 12 cm by 12 cm sheet by cutting a square of side x from each corner and turning up the flaps. What cut size x gives the largest volume?

SOLUTION

1. After cutting, the base is $(12 - 2x)$ by $(12 - 2x)$ and the height is x, so $V(x) = x(12 - 2x)^2$.

   set up the volume
2. Expand: $(12 - 2x)^2 = 144 - 48x + 4x^2$, so $V(x) = \boxed{144x - 48x^2 + 4x^3}$.
3. Differentiate: $V'(x) = \boxed{144 - 96x + 12x^2}$. *power rule, term by term*
4. Set to zero and divide by 12: $\boxed{x^2 - 8x + 12} = 0$, which factors as $(x - 2)(x - 6) = 0$, so $x = \boxed{2}$ or $x = \boxed{6}$. *set V'=0*
5. But $x = 6$ leaves no base (the flaps meet), so only $x = 2$ is feasible. Check: $V''(x) = 24x - 96$, and at $x = 2$, $V'' = 48 - 96 = -48 < 0$, a downward cup – a **maximum**.
6. Volume: $V(2) = 2(12 - 4)^2 = 2 \cdot 64 = \boxed{128}$ cm³.

Cut $x = \boxed{2}$ cm from each corner for the biggest box, holding $\boxed{128}$ cm³. The other root, $x = 6$, is a flat spot the geometry forbids.



Open-box volume from Example 9. The orange curve peaks at the feasible cut x=2 cm (volume 128 cm³); the other flat spot at x=6 collapses the base to nothing.

Note: Setting $f'(x) = 0$ finds the *candidates*, not the verdict. Always classify: a downward-cupped curve ($f'' < 0$) gives a maximum, an upward-cupped one ($f'' > 0$) gives a minimum. Skipping this step is the commonest optimisation mistake.

Mental-Math Shortcut

For any parabola $y = ax^2 + bx + c$, the single peak or valley sits at $x = -\frac{b}{2a}$ – which is just "$f'(x) = 0$" solved once and for all (since $f'(x) = 2ax + b = 0$). The sign of a tells the type instantly: $a > 0$ **opens up, so it is a minimum;** $a < 0$ **opens down, so it is a maximum.** And a handy fact worth memorising: whenever two numbers add to a fixed total, their *product* is largest when they are **equal** – so a fixed-perimeter rectangle is biggest as a square.

🧠 The Logic of It

Why must a maximum or minimum land *exactly* where the slope is zero – why can a peak never sit on a sloped piece of curve? The argument is a clean **proof by contradiction** (Chapter 25).

Suppose, for contradiction, that the true highest point of a smooth curve occurs at some $x = c$ where the slope is *not* zero – say the slope is positive there. A positive slope means the curve is still *rising* as we pass through c: points just to the right of c are higher than the value at c. But then c was not the highest point after all – we just found a taller one. That contradicts our assumption. The mirror argument rules out a negative slope (we would find a higher point just to the *left*). Either way we reach a contradiction.

The only escape is that the slope at a genuine peak is **neither positive nor negative** – it must be **zero**. The same reasoning, flipped, pins every valley to a zero slope too. So "$f'(x) = 0$" is not a lucky coincidence; logic *forces* it. (One honest caveat the logic also reveals: zero slope is *necessary*, not *sufficient* – a flat spot might be a peak, a valley, or a momentary level pause – which is precisely why step 3, the classify step, exists.)

⚠ Common Mistakes

- Stopping after solving $f'(x) = 0$. That only finds the flat spots – you must still classify each as a maximum or a minimum.
- Mixing up the second-derivative test: $f'' > 0$ is cupped *upward*, a **minimum**; $f'' < 0$ is cupped *downward*, a **maximum**. It is easy to flip these.
- Reporting the input x as the answer when the question asks for the maximum *value*. Plug x back into the original f to get the value (the height), not the location.
- Forgetting that the endpoints of a closed interval can also hold the largest or smallest value, even if the slope there is not zero.
- Assuming every flat tangent is a peak or valley. Some are level pauses where the curve flattens but keeps going the same way (an inflection).

Quiz

57.1.Q1. At a maximum or minimum of a smooth curve, what is the slope (the derivative)?

A. Positive

B. Negative

C. Zero

D. Undefined

57.1.Q2. What are the three steps to optimise a function?

57.1.Q3. Find the critical point of $f(x) = x^2 - 4x + 7$.

A. $x = 1$

B. $x = 2$

C. $x = 4$

D. $x = 7$

57.1.Q4. Using the second derivative, is $x = 2$ above a maximum or a minimum for $f(x) = x^2 - 4x + 7$?

57.1.Q5. Find where $f(x) = 6x - x^2$ is greatest, and its value there.

57.1.Q6. The second derivative at a critical point is $f''(x) = -5$. Is it a max or a min?

A. Maximum

B. Minimum

C. Neither

D. Cannot tell

57.1.Q7. You have 20 m of fence for a three-sided pen against a wall, with short sides x so $A = 20x - 2x^2$. What x maximises the area?

57.1.Q8. True or False: solving $f'(x) = 0$ by itself proves you have found a maximum.

A. True

B. False

🐍 From Concept to Code (Python)

Even without algebra, a computer can find a minimum by brute force: try a whole range of x values, compute the function at each, and keep the smallest. Scanning many closely-spaced points zeroes in on the bottom of the valley – exactly the spot where the slope would be zero.

```
def f(x):
    return x**2 - 4*x + 7            # a valley-shaped curve

best_x = None
best_y = None
x = -2.0
while x <= 6.0:                      # scan x from -2 up to 6 in tiny steps
    y = f(x)
    if best_y is None or y < best_y:
        best_x, best_y = x, y        # remember the lowest point so far
    x += 0.001

print("minimum near x =", round(best_x, 3))
print("minimum value  =", round(best_y, 3))
```

```
output
minimum near x = 2.0
minimum value  = 3.0
```

Try it:

1. Calculus says the minimum of $x^2 - 4x + 7$ is at $x = 2$ with value 3. Does the scan agree?
2. To find a *maximum* instead, change the comparison to `y > best_y` and try `f = lambda x: 6*x - x**2`. You should land near $x = 3$, value 9.
3. Make the step `0.01` instead of `0.001` and notice the answer is slightly less precise. A finer scan gives a sharper result – at the cost of more loops.

Answer Key – Chapter 57 – Optimisation

57.1.Q1. Zero – the tangent is flat at a peak or a valley.

57.1.Q2. Differentiate to get $f'(x)$; set $f'(x) = 0$ and solve for the critical points; classify each as a maximum or a minimum.

57.1.Q3. $x = 2$ – set $f'(x) = 2x - 4 = 0$ and solve.

57.1.Q4. A minimum – $f''(x) = 2 > 0$, an upward cup (and the value there is 3).

57.1.Q5. $f'(x) = 6 - 2x = 0$ gives $x = 3$; since $f''(x) = -2 < 0$ it is a maximum, with value $f(3) = 9$.

57.1.Q6. Maximum – a negative second derivative means a downward cup (a hill).

57.1.Q7. $x = 5$ m – set $A'(x) = 20 - 4x = 0$. The pen is 5×10 m with area 50 m².

57.1.Q8. False – it only finds a flat spot. You must classify it (it could be a maximum, minimum, or a level pause).

Chapter 58 · Integrals

How to add up infinitely many tiny pieces to measure the exact area trapped under a curve – and how that "summing" turns out to be the reverse of taking a derivative.

The Idea Behind It

Area under a curve is found by reversing a derivative: if $A(x)$ is the area so far and $F' = f$, then $A(x) = F(x)$. *but why?*

Let $A(x)$ be the area under $y = f(x)$ accumulated from the left edge up to the position x. Watch how that area *grows* as you nudge x a tiny bit further right, by a width h. The new sliver added is a thin strip of width h standing on the curve, and the rule we lean on is the area of that near-rectangle, **height times width**:

$$\boxed{A(x+h) - A(x) \approx f(x) \cdot h}$$

The strip has height about $f(x)$ (the curve barely changes across so narrow a width) and width h. Divide both sides by h to get the *rate* at which area is accumulating:

$$\frac{A(x+h) - A(x)}{h} \approx f(x)$$

The left side is exactly the difference quotient for A. Let the strip get infinitely thin, $h \to 0$, and the approximation becomes exact: the area grows at precisely the rate of the height.

$$A'(x) = \lim_{h \to 0} \frac{A(x+h) - A(x)}{h} = f(x)$$

So the area function A is an *antiderivative* of f: differentiating it gives f back. Differentiation and integration are reverse operations, which is the Fundamental Theorem.

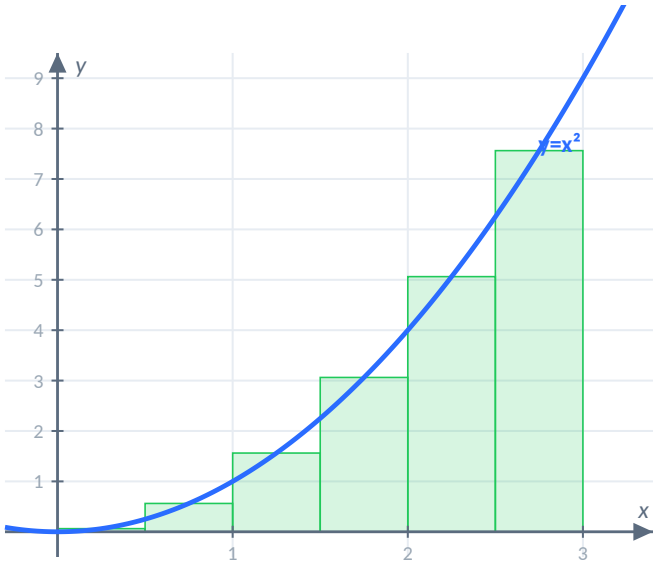
$$\therefore \int_a^b f(x)\,dx = F(b) - F(a), \quad \text{where } F' = f$$

This is why you never sum infinitely many rectangles by hand: just undo a derivative, then take the change across the interval. *area grows at the rate of the height*

A **derivative** chops a curve up to find its slope. An **integral** does the opposite job: it adds a curve up to find the *area underneath it*. That sounds like a geometry puzzle, but area-under-a-curve secretly measures real things – the distance a car travels from its speed, the total water that flows from a changing flow-rate, the energy stored from a varying force. This chapter builds the integral from one humble idea: slice the area into thin rectangles and add them.

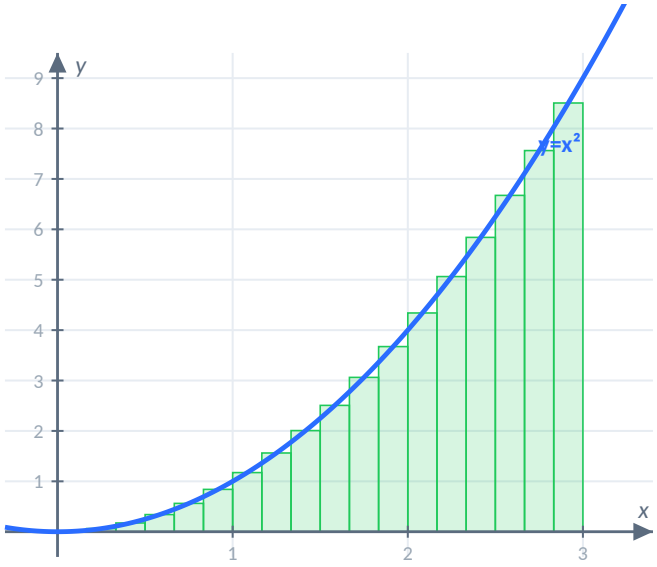
Area as a pile of rectangles

Take the curve $y = x^2$ between $x = 0$ and $x = 3$. The region under it has a curved top, so there is no neat length-times-width formula. But we can *approximate* it. Cut the base into a few strips, and on each strip stand a rectangle whose height is the curve. Add the rectangle areas and you get a rough estimate of the true area.



Six green rectangles approximate the area under y=x² from 0 to 3. Each rectangle's height is the curve at its midpoint.

The estimate is rough because the rectangles do not match the curve exactly – they overshoot in some places and undershoot in others. The fix is obvious: use *more, thinner* rectangles. The thinner they are, the better their tops trace the curve, and the closer the total gets to the true area.



Eighteen rectangles hug the curve far more tightly. As the count grows, the gap between rectangles and curve vanishes.

INTEGRAL

The **integral** of a function f from a to b, written $\int_a^b f(x)\,dx$, is the exact area between the curve $y = f(x)$ and the x-axis from $x = a$ to $x = b$. It is the limit of the rectangle total as the rectangles become infinitely thin.

The stretched "S" symbol $\int$ is an old long-form **S**, standing for **S**um. The pieces being summed are written $f(x)\,dx$: a height $f(x)$ times an infinitely small width dx. So $\int_a^b f(x)\,dx$ literally reads "sum, from a to b, of height-times-tiny-width". An integral is just a very careful sum.

> **Note:** The numbers a and b sitting on the integral sign are the **limits of integration** – the left and right edges of the region. They have nothing to do with the limits of Chapter 54 beyond sharing a name.

The shortcut: an integral undoes a derivative

Adding infinitely many rectangles by hand is hopeless. The miracle of calculus is that you almost never have to. To find the area, you look for an **antiderivative**: a function whose derivative is the one you started with. Run differentiation backwards.

> **ANTIDERIVATIVE**
>
> An **antiderivative** of $f(x)$ is a function $F(x)$ whose derivative is $f(x)$; that is, $F'(x) = f(x)$. Reversing the power rule: to undo $\frac{d}{dx}$, *raise the exponent by one and divide by the new exponent.*

$$\int x^n\,dx = \frac{x^{n+1}}{n+1} \quad \text{(up one, then divide)}$$

Check it works: differentiate $\frac{x^3}{3}$ and the power rule gives $\frac{3x^2}{3} = x^2$, right back to the start. So $\frac{x^3}{3}$ is an antiderivative of x^2. The link between *area* (summing rectangles) and *antiderivatives* (reversing slopes) is the crown jewel of the subject.

> **FUNDAMENTAL THEOREM OF CALCULUS (IN WORDS)**
>
> To find the exact area under f from a to b, find any antiderivative F, then subtract its value at the start from its value at the end: $\int_a^b f(x)\,dx = F(b) - F(a)$. In short: **the area is the change in the antiderivative across the interval.**

Worked examples

Example 1

Find an antiderivative of $f(x) = x^2$.

SOLUTION

1. Reverse the power rule: raise the exponent by one, $x^2 \to x^3$. *up one*
2. Divide by the new exponent: $\frac{x^3}{3}$. *then divide*
3. Check by differentiating: $\frac{d}{dx}\frac{x^3}{3} = \frac{3x^2}{3} = x^2$. Correct.

$F(x) = \frac{x^3}{3}$.

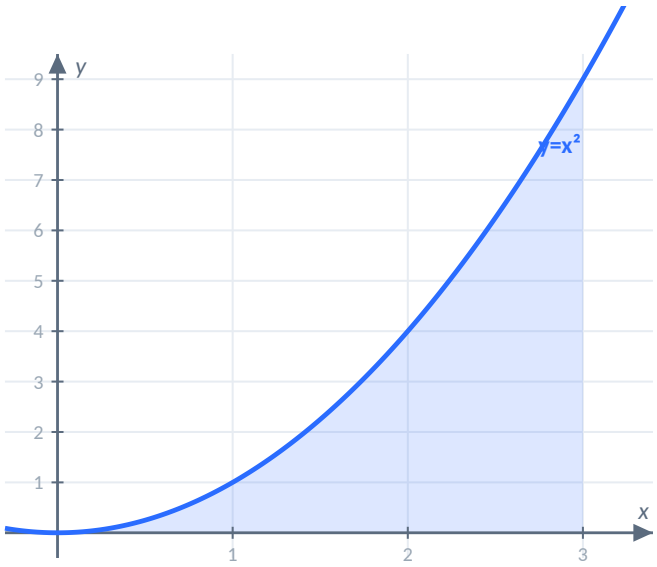
Example 2

Compute the exact area under $y = x^2$ from $x = 0$ to $x = 3$.

SOLUTION

1. Use the antiderivative from Example 1: $F(x) = \frac{x^3}{3}$.
2. Fundamental Theorem: area $= F(3) - F(0)$. *end – start*
3. $F(3) = \frac{3^3}{3} = 9$, and $F(0) = 0$.
4. Subtract: $9 - 0 = 9$.

The exact area is 9 – the value our rectangles were creeping toward.



The exact shaded area under y=x² from 0 to 3 is 9 square units.

Example 3

Find $\int_0^4 x\,dx$, and confirm with simple geometry.

SOLUTION

1. Antiderivative of $x = x^1$: up one to x^2, divide by 2, giving $\frac{x^2}{2}$. *up one, divide*
2. Evaluate: $\frac{4^2}{2} - \frac{0^2}{2} = \frac{16}{2} - 0 = 8$. *end – start*
3. Geometry check: under $y = x$ from 0 to 4 is a triangle, base 4, height 4, area $\frac{1}{2} \cdot 4 \cdot 4 = 8$. Agrees.

8.

Example 4

Find $\int_1^2 (2x + 1)\,dx$.

SOLUTION

1. Integrate term by term. $2x \to x^2$, and $1 \to x$. *up one, divide*
2. So an antiderivative is $F(x) = x^2 + x$.
3. Evaluate at the ends: $F(2) = 2^2 + 2 = 6$ and $F(1) = 1^2 + 1 = 2$.
4. Subtract: $6 - 2 = 4$. *end – start*

4.

Example 5

A car's speed is $v(t) = 2t$ metres per second. How far does it travel between $t = 0$ and $t = 5$ seconds?

SOLUTION

1. Distance is the area under the speed–time graph, so integrate $v(t)$.
2. Antiderivative of $2t$ is t^2. *reverse the derivative*
3. Evaluate: $5^2 - 0^2 = 25$. *end – start*

It travels 25 metres – the area under its speed graph.

Example 6

Estimate $\int_0^2 x^2\,dx$ using just two rectangles of width 1, taking the height at each strip's midpoint.

SOLUTION

1. Strip 1 spans $x = 0$ to 1; its midpoint is 0.5, height $0.5^2 = 0.25$. Area $= 1 \times 0.25 = 0.25$.
   height × width
2. Strip 2 spans $x = 1$ to 2; its midpoint is 1.5, height $1.5^2 = 2.25$. Area $= 1 \times 2.25 = 2.25$.
3. Add the two rectangles: $0.25 + 2.25 = 2.5$. *sum the strips*
4. The exact value is $\frac{2^3}{3} = \frac{8}{3} \approx 2.67$, so two rectangles already land close.

Estimate ≈ 2.5; the true area is $\frac{8}{3} \approx 2.67$.

Example 7

Evaluate $\int_1^2 (3x^2 - 2x + 1)\,dx$.

SOLUTION

1. Integrate term by term, raising each exponent by one and dividing. $3x^2 \to \frac{3x^3}{3} = x^3$. *up one, divide*
2. $-2x \to \frac{-2x^2}{2} = -x^2$, and $1 \to x$.
3. So an antiderivative is $F(x) = x^3 - x^2 + x$.
4. Evaluate at the ends: $F(2) = 8 - 4 + 2 = 6$ and $F(1) = 1 - 1 + 1 = 1$.
5. Subtract: $6 - 1 = 5$. *end − start*

$$\int_1^2 (3x^2 - 2x + 1)\,dx = 5.$$

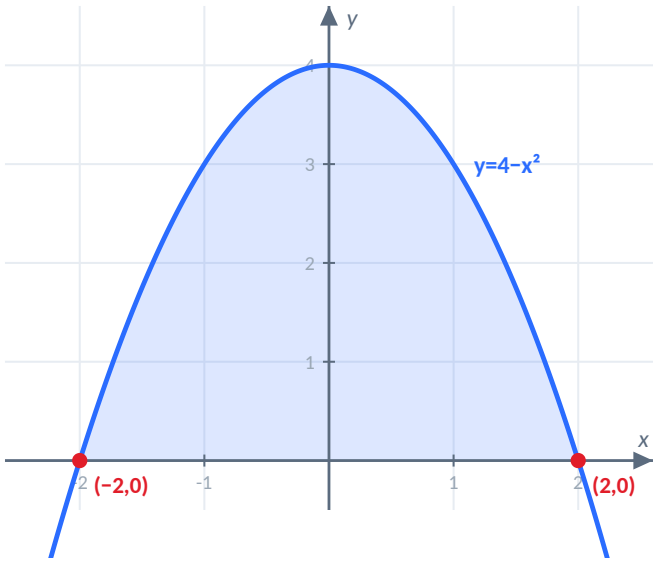
Example 8

Find the exact area trapped between the curve $y = 4 - x^2$ and the x-axis, where the curve sits above the axis.

SOLUTION

1. First locate the edges: the curve meets the axis where $4 - x^2 = 0$, i.e. $x^2 = 4$, so $x = -2$ and $x = 2$. Between them the curve is above the axis. *find the crossings*
2. The area is $\int_{-2}^{2} (4 - x^2)\,dx$. Antiderivative: $4 - x^2 \to F(x) = 4x - \frac{x^3}{3}$. *up one, divide*
3. At the right edge: $F(2) = 8 - \frac{8}{3} = \frac{24 - 8}{3} = \frac{16}{3}$.
4. At the left edge: $F(-2) = -8 - \frac{-8}{3} = -8 + \frac{8}{3} = -\frac{16}{3}$.
5. Subtract: $\frac{16}{3} - \left(-\frac{16}{3}\right) = \frac{32}{3} = \frac{32}{3}$. *end – start*

The enclosed area is $\frac{32}{3} \approx 10.67$ square units – the whole shaded dome below.



The shaded dome under y=4−x² between its two roots x=−2 and x=2 has exact area 32/3 ≈ 10.67, from Example 8.

Example 9

Estimate $\int_0^4 x^2\,dx$ with four rectangles of width 1, taking the height at each strip's **midpoint**, then compare with the exact value.

SOLUTION

1. The four strips span $[0,1], [1,2], [2,3], [3,4]$; their midpoints are $0.5,\ 1.5,\ 2.5,\ 3.5$.

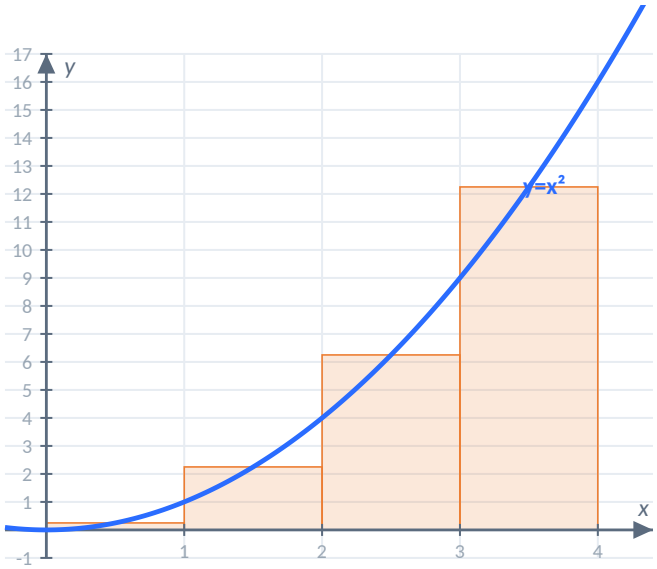
   midpoint of each strip

2. Heights are the curve there: $0.5^2,\ 1.5^2,\ 2.5^2,\ 3.5^2 = 0.25,\ 2.25,\ 6.25,\ 12.25$.
3. Each width is 1, so the rectangle areas equal those heights. Add them: $0.25 + 2.25 + 6.25 + 12.25 = 21$.

   sum the strips

4. The exact value is $\frac{4^3}{3} = \frac{64}{3} \approx 21.33$, so just four midpoint rectangles already land within a third of a unit.

Midpoint estimate $= 21$ against the exact $\frac{64}{3} \approx 21.33$ – midpoints hug a curve far better than corner heights.



Four midpoint rectangles under y=x² from 0 to 4 (Example 9). Each top meets the curve at the strip's centre, so the total 21 already sits close to the exact area 64/3 ≈ 21.33.

Notice the pattern in Examples 2–5: the answer is always "antiderivative at the right edge minus antiderivative at the left edge". Once you can reverse a derivative, you can measure an area – and an area can stand for distance, total, or accumulation in the real world.

⚡ Mental-Math Shortcut

Integration is the power rule **run backwards**: *"up and divide."* Where differentiating brought the exponent *down* and *reduced* it, integrating raises the exponent *up* by one and *divides* by that new exponent. So $x^2 \to \frac{x^3}{3}$, $x^4 \to \frac{x^5}{5}$, and a plain $x \to \frac{x^2}{2}$. Differentiate is "down and down"; integrate is "up and divide" – exact mirror images.

Why does a *sum of rectangles* settle on one exact number instead of wandering forever? It is a squeeze argument, the same shape as Chapter 55's limit. Trap the true area A under $y = x^2$ on $[0, 1]$ from **below** and from **above** using n equal strips.

The thin rectangles that stay *under* the curve total a little less than A; the ones that poke *above* total a little more. A short calculation gives a lower estimate $L_n = \frac{(n-1)n(2n-1)}{6n^3}$ and an upper estimate $U_n = \frac{n(n+1)(2n+1)}{6n^3}$, with the true area pinned between them:

$$L_n \le A \le U_n.$$

Now reason it through. Their *difference* is $U_n - L_n = \frac{1}{n}$ (the area of the single end strip). As n grows, that gap shrinks toward 0, so the lower and upper walls are squeezed together. Both L_n and U_n march toward $\frac{1}{3}$. Since A is *forever* caught between two values that both approach $\frac{1}{3}$, the only number it can equal is $\frac{1}{3}$. That "trapped between, gap closing, therefore forced" structure is exactly why an integral is a single, definite number – and indeed $\int_0^1 x^2\,dx = \frac{1^3}{3} = \frac{1}{3}$.

⚠ Common Mistakes

- Integrating like you differentiate. Differentiating x^2 gives $2x$; **integrating** it gives $\frac{x^3}{3}$. The arrows point opposite ways.
- Forgetting to divide by the new exponent: $\int x^2\,dx = \frac{x^3}{3}$, **not** x^3.
- Subtracting the wrong way round. The area is $F(b) - F(a)$ (end minus start), not $F(a) - F(b)$.
- Thinking a few fat rectangles give the exact area. They only *approximate* it; the exact value is the limit as the rectangles become infinitely thin.
- Reading $\int f(x)\,dx$ as multiplication of f, x and d. It is one symbol meaning "sum the tiny pieces $f(x)\,dx$".

Quiz

58.1.Q1. In plain words, what does the integral $\int_a^b f(x)\,dx$ measure?

58.1.Q2. What is an antiderivative of $f(x) = x^3$?

A. $\frac{x^4}{4}$

B. $3x^2$

C. x^4

D. $\frac{x^2}{2}$

58.1.Q3. Find $\int_0^2 x\,dx$.

58.1.Q4. Why do we use more rectangles when approximating an area?

A. Thinner rectangles hug the curve more closely, so the total is more accurate

B. To make the picture prettier

C. It makes the integral smaller

D. More rectangles change the true area

58.1.Q5. The Fundamental Theorem says the exact area equals which of these?

A. $F(b) - F(a)$

B. $F(a) - F(b)$

C. $F(b) \times F(a)$

D. $f(b) - f(a)$

58.1.Q6. A car's speed is $v(t) = 2t$. What does the area under its speed–time graph represent?

58.1.Q7. Compute $\int_1^3 2x\,dx$.

58.1.Q8. True or False: differentiating then integrating are reverse operations.

A. True

B. False

🐍 From Concept to Code (Python)

A computer finds an integral the *direct* way: chop the base into many narrow rectangles, add their areas, and let the strips be very thin. This is the picture above turned into a loop – no algebra needed.

```
def integral(f, a, b, n=10000):
    dx = (b - a) / n                # width of each thin strip
    total = 0.0
    for i in range(n):
        x = a + (i + 0.5) * dx      # height at the strip's midpoint
        total += f(x) * dx          # area of this rectangle
    return total

f = lambda x: x**2                  # the curve y = x²
print(round(integral(f, 0, 3), 4))  # area from 0 to 3
print(round(integral(f, 0, 2), 4))  # ...and from 0 to 2
```

```
output
9.0
2.6667
```

Try it:

1. The exact value of $\int_0^3 x^2\,dx$ is 9. Lower `n` to `4` and watch the answer drift away from 9 with only a few fat rectangles.
2. Change `f` to `lambda x: 2*x` and confirm $\int_0^5 2x\,dx$ is close to 25 (the car in Example 5).
3. Use it to estimate $\int_0^1 x**3\,dx$ and check it is near $\frac{1}{4} = 0.25$.

Answer Key – Chapter 58 – Integrals

58.1.Q1. The area between the curve $y = f(x)$ and the x-axis, from $x = a$ to $x = b$.

58.1.Q2. $\frac{x^4}{4}$ – raise the exponent to 4 and divide by 4; differentiating it returns x^3.

58.1.Q3. 2 – the antiderivative is $\frac{x^2}{2}$, and $\frac{2^2}{2} - 0 = 2$ (a triangle of base 2, height 2).

58.1.Q4. Thinner rectangles hug the curve more closely, so the total approaches the exact area.

58.1.Q5. $F(b) - F(a)$ – the antiderivative at the end minus at the start.

58.1.Q6. The distance travelled – integrating speed over time gives distance.

58.1.Q7. 8 – antiderivative x^2, so $3^2 - 1^2 = 9 - 1 = 8$.

58.1.Q8. True – an integral undoes a derivative, which is the heart of the Fundamental Theorem.

Chapter 59 · Multivariable Calculus: Partial Derivatives & Gradients

How to measure the slope of a surface that depends on two things at once – by changing one variable at a time – and how stacking those slopes into a vector points straight uphill.

The Idea Behind It

The gradient ∇f points in the direction of *steepest ascent* on a surface. but why?

Stand on a hillside and pick a compass direction to walk, a unit arrow $\mathbf{u}$ of length 1. Some directions climb, some descend; we want the steepest climb. The fact we lean on is the formula for the slope you feel walking along $\mathbf{u}$, the **directional derivative**, written with a dot product:

$$\boxed{\text{slope along } \mathbf{u} = \nabla f \cdot \mathbf{u} = |\nabla f|\,|\mathbf{u}|\,\cos\theta}$$

Here θ is the angle between your heading $\mathbf{u}$ and the gradient. Since $\mathbf{u}$ has length 1 and $|\nabla f|$ is fixed by where you stand, only $\cos\theta$ can vary:

$$\text{slope along } \mathbf{u} = |\nabla f|\,\cos\theta$$

A cosine is largest when it equals 1, and $\cos\theta = 1$ exactly when $\theta = 0$, that is, when you walk *along* the gradient itself. No other heading can give a bigger slope:

$$\therefore\ \text{the slope is greatest when } \theta = 0,\ \text{i.e. heading along } \nabla f$$

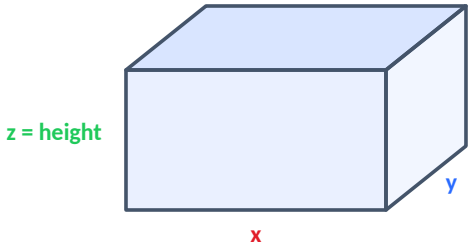
So the gradient is the "straight uphill" compass, and its length $|\nabla f|$ is how steep that steepest climb is. The exact opposite, $\cos\theta = -1$ at $\theta = 180^\circ$, is steepest *descent*, which is why gradient descent steps along $-\nabla f$.

biggest when you face the gradient

So far a function took one input and gave one output: $y = f(x)$. But the real world rarely depends on just one thing. The temperature in a room depends on *where you are*: an east–west position x **and** a north–south position y. The height of a hillside depends on two map coordinates. We write such a thing $z = f(x, y)$: feed in two numbers, get one out. Its graph is no longer a curve but a **surface** – a landscape of hills and valleys floating above the flat xy-plane.

FUNCTION OF TWO VARIABLES

A rule $z = f(x, y)$ that takes a pair of inputs (x, y) and returns a single output z. Picture x and y as a position on a map and z as the height of the ground there.



A function z = f(x,y) is a surface above the flat xy-floor: pick a spot (x,y) and z is the height there.

One slope is not enough

On a hillside, "how steep is it here?" has no single answer – *it depends which way you face.* Walk east and the ground may climb; turn and walk north and it may fall. A surface has many slopes at one point, one for each direction. To get a grip, we measure the two simplest ones: the slope as you step purely east (along x) and the slope as you step purely north (along y).

PARTIAL DERIVATIVE

The **partial derivative** of $f(x, y)$ with respect to x, written $\frac{\partial f}{\partial x}$, is its slope in the x-direction *while y is frozen*. To compute it, treat y as a constant number and differentiate as usual. Likewise $\frac{\partial f}{\partial y}$ freezes x.

Note: The rounded ∂ ("partial") replaces the straight d to remind you that other variables are being held still. Computing a partial derivative is ordinary differentiation – you just pretend the other letter is a frozen constant like 7.

Geometrically, freezing y means slicing the surface with a vertical wall running east–west. The cut edge is an ordinary curve, and $\frac{\partial f}{\partial x}$ is just its slope – the familiar tangent from Chapter 55, applied to one slice.

Worked examples

Example 1

For $f(x, y) = x^2 + y^2$, find both partial derivatives.

SOLUTION

1. For $\frac{\partial f}{\partial x}$, freeze y: the term y^2 is now a constant, so its slope is 0. Differentiate x^2 normally: $2x$.
   freeze y
2. For $\frac{\partial f}{\partial y}$, freeze x: now x^2 is the constant (slope 0), and $y^2 \to 2y$. *freeze x*

$\frac{\partial f}{\partial x} = 2x$ **and** $\frac{\partial f}{\partial y} = 2y$.

Example 2

For $f(x, y) = 3x^2 + 5y$, find both partials.

SOLUTION

1. $\frac{\partial f}{\partial x}$: hold y fixed, so $5y$ is constant (slope 0); $3x^2 \to 6x$. *freeze y*
2. $\frac{\partial f}{\partial y}$: hold x fixed, so $3x^2$ is constant (slope 0); $5y \to 5$. *freeze x*

$\frac{\partial f}{\partial x} = 6x$ **and** $\frac{\partial f}{\partial y} = 5$.

Example 3

For $f(x, y) = xy$, find both partials.

SOLUTION

1. $\frac{\partial f}{\partial x}$: freeze y, so y is just a constant multiplier on x. The slope of $(\text{constant}) \cdot x$ is that constant: y.
   y is a coefficient
2. $\frac{\partial f}{\partial y}$: freeze x, so x multiplies y; the slope is x.

$\frac{\partial f}{\partial x} = y$ **and** $\frac{\partial f}{\partial y} = x$.

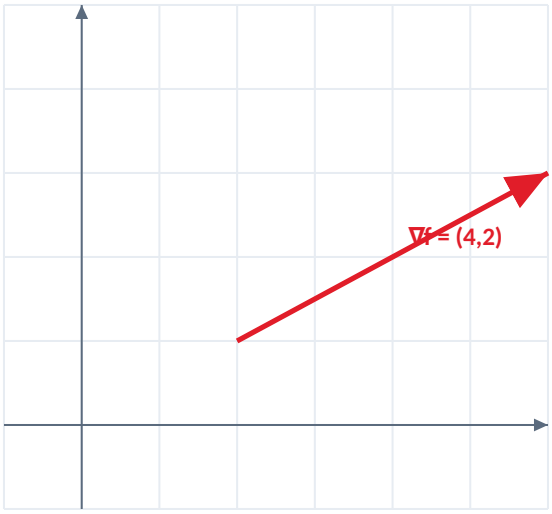
The gradient: stack the slopes into an arrow

Two partial derivatives, one per direction. Bundle them into a single object – a **vector** – and something beautiful happens: the arrow it forms points in the direction the surface climbs *fastest*.

GRADIENT

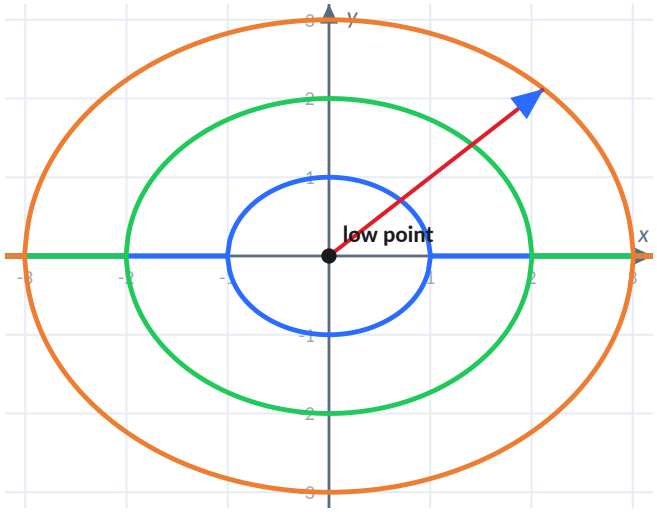
The **gradient** of f, written ∇f (say "grad f"), is the vector of its partial derivatives: $\nabla f = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)$. At any point it **points in the direction of steepest ascent**, and its length tells you how steep that steepest climb is.

Think of standing on a hill and asking "which way is straight uphill?" The gradient is the compass that answers. For $f(x, y) = x^2 + y^2$ – a bowl with its low point at the origin – the gradient is $(2x,\ 2y)$. At the point $(2, 1)$ it is $(4, 2)$, an arrow pointing up-and-to-the-right, away from the centre of the bowl, i.e. uphill out of the valley.



The gradient of f = x²+y² at the point (2,1) is the red arrow (4,2): it points directly uphill, away from the bowl's low centre.

We can also picture a surface from directly above using a **contour map**, exactly like a hiking map. Each ring joins points of equal height. For the bowl $x^2 + y^2$, points at the same height sit on circles around the origin – tight rings where it is steep, wide rings where it is gentle. The gradient always crosses these rings at right angles, pointing toward the higher ones.



Contour map of the bowl $z = x^2+y^2$ seen from above: each ring is one height. The red gradient arrow crosses the rings at right angles, pointing toward higher ground.

Example 4

For $f(x, y) = x^2 + y^2$, find the gradient at the point $(3, 4)$ and say which way it points.

SOLUTION

1. The partials are $\frac{\partial f}{\partial x} = 2x$ and $\frac{\partial f}{\partial y} = 2y$ (from Example 1).
2. At $(3, 4)$: $2x = 6$ and $2y = 8$. *plug in the point*
3. So $\nabla f = (6, 8)$, an arrow pointing up-and-right, directly away from the origin. *stack into a vector*

$\nabla f(3, 4) = (6, 8)$ **– steepest climb is in that direction.**

Example 5

At the very bottom of the bowl, the origin $(0, 0)$, what is the gradient – and what does it mean?

SOLUTION

1. Plug into $\nabla f = (2x, 2y)$: $(2 \cdot 0,\ 2 \cdot 0) = (0, 0)$. *plug in the point*
2. A zero vector has no length and no direction.
3. No direction is uphill because every direction climbs equally from the lowest point – the ground is momentarily flat. *flat spot = min*

$\nabla f = (0, 0)$ **: a flat spot, which signals a minimum (the bottom of the bowl). Zero gradient marks the peaks, pits and passes of a surface.**

Example 6

A surface mixes its variables: $f(x, y) = x^2y + y^3$. Find both partial derivatives.

SOLUTION

1. For $\frac{\partial f}{\partial x}$, freeze y. In x^2y the frozen y is just a constant multiplier, and $x^2 \to 2x$, giving $2xy$. The term y^3 has no x, so it vanishes to 0. *freeze y; y^3 has no x*
2. For $\frac{\partial f}{\partial y}$, freeze x. Now x^2y is (constant x^2) times y, whose slope is that constant: x^2. And $y^3 \to 3y^2$.
   freeze x; x^2 is the coefficient
3. Add each term back together within its own partial.

$\frac{\partial f}{\partial x} = 2xy$ **and** $\frac{\partial f}{\partial y} = x^2 + 3y^2$.

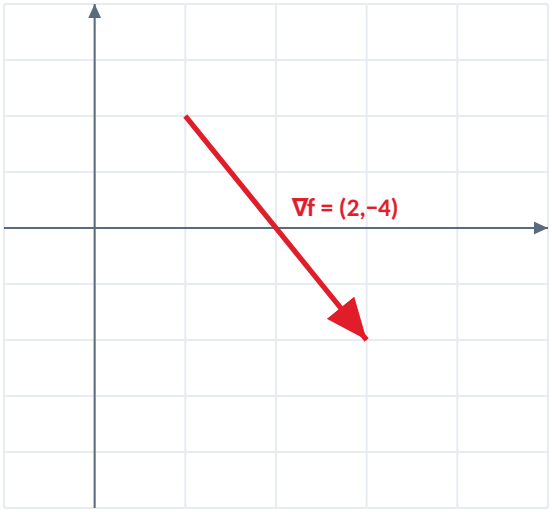
Example 7

For the saddle-shaped surface $f(x, y) = x^2 - y^2$, find the gradient at $(1, 2)$ and draw it.

SOLUTION

1. Partial derivatives: freezing y gives $\frac{\partial f}{\partial x} = 2x$; freezing x gives $\frac{\partial f}{\partial y} = -2y$ (note the minus on $-y^2$).
   mind the sign
2. At $(1, 2)$: $2x = 2(1) = 2$ and $-2y = -2(2) = -4$. *plug in the point*
3. Stack into a vector: $\nabla f = (2, -4)$ – it points right and *downward*, since climbing in x raises the surface but climbing in y lowers it. *up in x, down in y*

$\nabla f(1, 2) = (2, -4)$ **– the steepest-ascent arrow tips downward here because the y-direction runs downhill on a saddle.**



The gradient of the saddle $f = x^2 - y^2$ at (1,2) is the red arrow (2,−4): steepest ascent heads right-and-down, because moving in y descends the saddle (Example 7).

Example 8

You stand at $(3, 4)$ on the bowl $f(x, y) = x^2 + y^2$. Which direction takes you downhill fastest, and how steep is that descent?

SOLUTION

1. The gradient there is $\nabla f = (2x, 2y) = (2 \cdot 3,\ 2 \cdot 4) = \boxed{(6, 8)}$, pointing steepest *uphill*.

   gradient = steepest up

2. Steepest descent is the exact opposite direction: $-\nabla f = \boxed{(-6, -8)}$ – back toward the centre of the bowl.

   flip the sign for downhill

3. How steep? The slope along the gradient is its length: $|\nabla f| = \sqrt{6^2 + 8^2} = \sqrt{36 + 64} = \sqrt{100} = \boxed{10}$.

   length = steepness

Fastest descent points along $\boxed{(-6, -8)}$, and the ground there falls at a rate of $\boxed{10}$ – this "step opposite the gradient" is one move of gradient descent.

Tip: A zero gradient is the multivariable version of "slope = 0" from optimisation. To find the highest or lowest point of a surface, hunt for where *both* partial derivatives vanish at once.

This is why gradients run modern computing. Training a neural network means tuning millions of numbers to make an error as small as possible. The machine computes the gradient of the error and steps in the *opposite* direction – downhill – over and over. That single trick, "follow the gradient down", is called gradient descent, and it is partial derivatives doing all the work.

⚡ Mental-Math Shortcut

To take a partial derivative, **freeze the other letter and pretend it is a number.** Want $\frac{\partial}{\partial x}$? Mentally swap every y for a frozen constant like 7, differentiate in x the usual way, then put y back. Any term with no x in it is constant, so it just vanishes to 0. Then do the mirror image for $\frac{\partial}{\partial y}$. Two ordinary derivatives, taken one variable at a time.

Why must the gradient point along the *steepest* uphill direction, and not some other way? It follows from a clean logical chain about how each direction's slope is built.

Stand at a point and choose a direction to walk, described by a unit arrow $\mathbf{u} = (u_1, u_2)$ (length 1). A theorem of multivariable calculus says the slope you feel walking that way, the **directional slope**, is the gradient combined with your direction:

$$\text{slope in direction } \mathbf{u} = \nabla f \cdot \mathbf{u} = |\nabla f| \, \cos\theta,$$

where θ is the angle between your heading and the gradient. Now reason it through, one forced step at a time. The only part you control is θ; the length $|\nabla f|$ is fixed by where you stand. The slope is largest exactly when $\cos\theta$ is largest. And $\cos\theta$ is at its maximum value of 1 precisely when $\theta = 0$ – that is, when you walk *along* the gradient. *Therefore* no direction is steeper than the gradient's own. By the same logic, $\cos\theta = -1$ at $\theta = 180^\circ$, so the steepest *descent* is the exact opposite direction. The "biggest when aligned" structure is the same maximising move from Chapter 57's optimisation, now lifted into two dimensions.

⚠ Common Mistakes

- Differentiating both letters at once. A partial derivative changes **one** variable while the other is held frozen.
- Letting the frozen variable disappear when it is a *multiplier*. In $f = xy$, holding y fixed gives $\frac{\partial f}{\partial x} = y$, not 0 – y is a constant **coefficient**, not a vanishing term.
- Treating the gradient as a single number. It is a **vector**: one component per variable, carrying a direction.
- Thinking the gradient points along a contour line. It points *across* the rings, at right angles, toward higher ground.
- Forgetting that a zero gradient flags a flat spot – a peak, pit or pass – not "nothing to see here".

Quiz

59.1.Q1. What does $z = f(x, y)$ describe geometrically?

A. A surface (a landscape of height z above the xy-plane)

B. A single straight line

C. A flat circle

D. A number

59.1.Q2. To find $\frac{\partial f}{\partial x}$, what do you do with y?

59.1.Q3. For $f(x, y) = x^2 + y^2$, find $\frac{\partial f}{\partial y}$.

A. $2y$

B. $2x$

C. x^2

D. 0

59.1.Q4. For $f(x, y) = xy$, what is $\frac{\partial f}{\partial x}$?

59.1.Q5. What is the gradient ∇f?

A. The vector of partial derivatives, pointing uphill

B. The area under the surface

C. A single average slope

D. The lowest point

59.1.Q6. For $f = x^2 + y^2$, compute the gradient at $(1, 2)$.

59.1.Q7. What does a gradient of $(0, 0)$ tell you?

59.1.Q8. True or False: to go downhill fastest, step in the direction opposite the gradient.

A. True

B. False

🐍 From Concept to Code (Python)

Just as a one-variable slope can be measured with a tiny step, a partial derivative is the average slope over a tiny step in **one** coordinate while the other stays put. Nudge x for $\partial f/\partial x$; nudge y for $\partial f/\partial y$.

```
def dfdx(f, x, y, h=1e-6):
    return (f(x + h, y) - f(x, y)) / h   # nudge x only

def dfdy(f, x, y, h=1e-6):
    return (f(x, y + h) - f(x, y)) / h   # nudge y only

f = lambda x, y: x**2 + y**2             # the bowl z = x² + y²

# gradient at the point (3, 4): exact answer is (6, 8)
print(round(dfdx(f, 3, 4), 3), round(dfdy(f, 3, 4), 3))
```

```
output
6.0 8.0
```

Try it:

1. Check the gradient at the origin $(0, 0)$ is close to $(0, 0)$ – the flat bottom of the bowl.
2. Swap in `f = lambda x, y: x*y` and confirm at $(3, 4)$ you get about $(4, 3)$, matching $\dfrac{\partial f}{\partial x} = y$, $\dfrac{\partial f}{\partial y} = x$.
3. The length of the gradient is `math.hypot(dfdx(...), dfdy(...))`. Compute it at $(3, 4)$ for the bowl and see it equals 10.

Answer Key – Chapter 59 – Multivariable Calculus: Partial Derivatives & Gradients

59.1.Q1. A surface – height z above each map position (x,y).

59.1.Q2. Hold y fixed – treat it as a constant – and differentiate with respect to x.

59.1.Q3. $2y$ – freeze x (so x^2 is constant) and differentiate y^2.

59.1.Q4. y – with y frozen it is a constant multiplier on x, and the slope of (constant)·x is that constant.

59.1.Q5. The vector of the partial derivatives; it points in the direction of steepest ascent.

59.1.Q6. $(2, 4)$ – plug into $\nabla f = (2x, 2y)$.

59.1.Q7. You are at a flat spot – a maximum, minimum or saddle point (here, the bottom of the bowl).

59.1.Q8. True – the gradient points steepest uphill, so its opposite is steepest downhill (the basis of gradient descent).

Chapter 60 · Differential Equations

How to write down a rule for the rate at which something changes, and then recover the whole story – the function over time – that obeys that rule.

The Idea Behind It

e^x is its own derivative, which is exactly why it models growth: $\frac{d}{dx}e^x = e^x$. but why?

Among all the curves a^x, the slope is always some fixed multiple of the height. The number $e \approx 2.718$ is *defined* as the one base where that multiple is exactly 1, so the slope equals the height at every point. That single property is the fact we lean on:

$$\boxed{\frac{d}{dx}\, e^x = e^x}$$

Now test the candidate $y = e^{kt}$ against the growth rule "rate is proportional to amount", $\frac{dy}{dt} = k\,y$. Differentiate it; the chain rule brings the k down from the exponent:

$$\frac{d}{dt}\, e^{kt} = k\, e^{kt}$$

The right side is just k times the original e^{kt}, which is k times y. So the rule is satisfied at every instant, and a starting amount y_0 rides along as a constant multiplier:

$$\frac{dy}{dt} = k\, e^{kt} = k\, y$$

$$\therefore \quad \frac{dy}{dt} = k\, y \quad \text{is solved by} \quad y(t) = y_0\, e^{kt}$$

This is why "rate proportional to amount", a big population breeding fast, a hot drink shedding heat, money compounding, always produces an exponential. The self-reproducing slope of e is the engine of growth and decay. slope equals height, so growth feeds itself

An ordinary equation like $x + 3 = 7$ hides an unknown **number**. A **differential equation** hides an unknown **function**, and it describes that function indirectly – through its rate of change, its derivative. This is the natural language of nature. We rarely know a quantity outright, but we often know the *rule for how fast it changes*: a population grows in proportion to its size, a hot drink cools in proportion to how much hotter it is than the room, a savings balance grows in proportion to itself. Each of those sentences is a differential equation.

DIFFERENTIAL EQUATION

An equation that contains a derivative of an unknown function. Its **solution** is not a number but a whole **function** $y(t)$ whose rate of change matches the equation at every instant.

The most important one in all of science

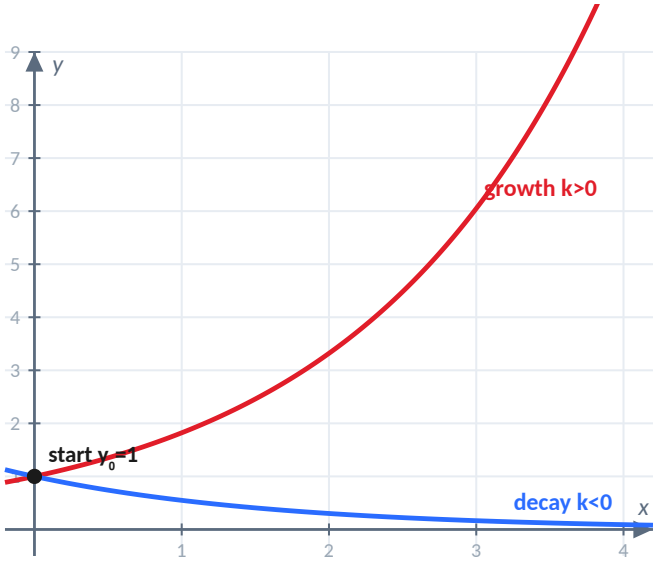
One differential equation appears everywhere. In words: *the rate of change of y is proportional to y itself.* In symbols:

$$\frac{dy}{dt} = k\,y$$

Read it as "the speed at which y grows depends on how much y there already is". A big population has many parents, so it grows fast; a small one grows slowly. The constant k sets the pace. The solution – the function that obeys this rule at every moment – is an **exponential**:

$$y(t) = y_0\,e^{kt}$$

Here y_0 is the starting amount (the value at $t = 0$) and $e \approx 2.718$ is the natural growth constant. If $k > 0$ the quantity **grows**, faster and faster; if $k < 0$ it **decays**, fading toward zero. We can confirm the solution fits: differentiating $y_0 e^{kt}$ gives $k\,y_0 e^{kt} = k\,y$, exactly the rule we started from.



Both curves start at y_0=1. With k>0 (red) y races upward; with k<0 (blue) it decays toward zero. Both obey dy/dt = ky.

Note: The bigger $|k|$ is, the more dramatic the curve – steeper growth or quicker decay. The sign of k alone decides the *direction*: positive climbs, negative fades.

Worked examples

Example 1

Verify that $y = e^{2t}$ solves the differential equation $\dfrac{dy}{dt} = 2y$.

SOLUTION

1. Differentiate the candidate: $\dfrac{d}{dt}\, e^{2t} = 2e^{2t}$ (the chain rule brings the 2 down). *chain rule*
2. Compare with the right-hand side $2y = 2e^{2t}$.
3. They are identical, so the rule holds at every t. *both sides match*

Yes – its rate of change really is 2 times itself, so $y = e^{2t}$ is a solution.

Example 2

A colony of bacteria grows by $\dfrac{dy}{dt} = 0.5\,y$, starting at $y_0 = 100$ cells. Write the solution and find the count at $t = 4$ hours.

SOLUTION

1. Match to $\dfrac{dy}{dt} = ky$: here $k = 0.5$ and $y_0 = 100$. *read off k and y0*
2. So $y(t) = 100\,e^{0.5t}$.
3. At $t = 4$: $y = 100\,e^{0.5\times 4} = 100\,e^{2}$. *plug in t=4*
4. Since $e^2 \approx 7.39$, the count is about 739.

$y(t) = 100e^{0.5t}$**, giving roughly** 739 **cells after 4 hours.**

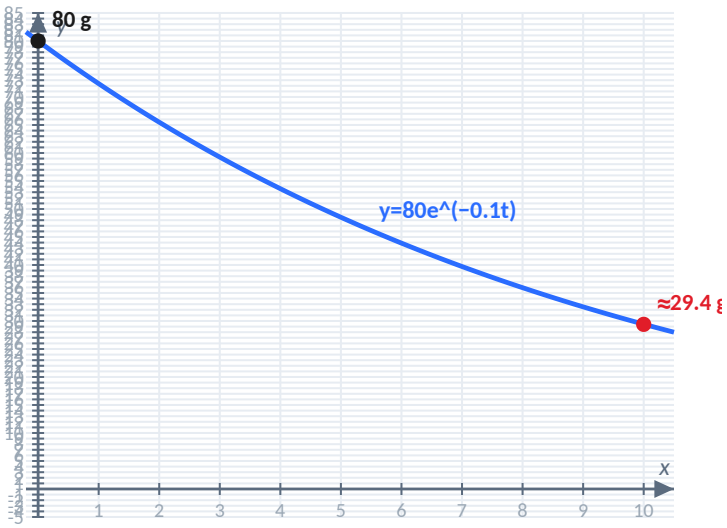
Example 3

A radioactive sample decays by $\dfrac{dy}{dt} = -0.1\,y$ from $y_0 = 80$ grams. How much remains at $t = 10$?

SOLUTION

1. Here $k = -0.1$ (negative, so decay) and $y_0 = 80$. *k<0 means decay*
2. Solution: $y(t) = 80\,e^{-0.1t}$.
3. At $t = 10$: $y = 80\,e^{-0.1\times 10} = 80\,e^{-1}$. *plug in t=10*
4. Since $e^{-1} \approx 0.368$, about 29.4 grams are left.

$y(t) = 80e^{-0.1t}$**, so about** 29.4 **grams remain after 10 units of time.**



Radioactive decay from Example 3: the sample falls from 80 g, never quite reaching zero – the hallmark of exponential decay.

Example 4

Without solving exactly, sketch the behaviour of $\frac{dy}{dt} = 3y$ starting from $y_0 = 2$. Is it growth or decay, and how fast initially?

SOLUTION

1. The constant $k = 3$ is positive, so this is **growth**. k>0 means grow
2. The initial rate is $\frac{dy}{dt} = 3y = 3 \times 2 = 6$: at $t = 0$ the quantity is climbing by 6 units per unit time.
3. As y rises, that rate rises too, so the curve gets ever steeper – classic runaway growth.

Fast accelerating growth; the solution is $y = 2e^{3t}$.

Marching forward in small steps: Euler's method

Most differential equations have no tidy formula like e^{kt}. So computers solve them by *crawling forward*. The derivative tells you the current rate; multiply it by a tiny time step Δt to guess how much y changes, then take the step. Repeat. This is **Euler's method**, and it is nothing more than the slope idea from Chapter 55 run in reverse: instead of finding a slope, we *use* the known slope to predict the next value.

$$y_{\text{next}} = y_{\text{now}} + \Delta t \cdot \left.\frac{dy}{dt}\right|_{\text{now}}$$

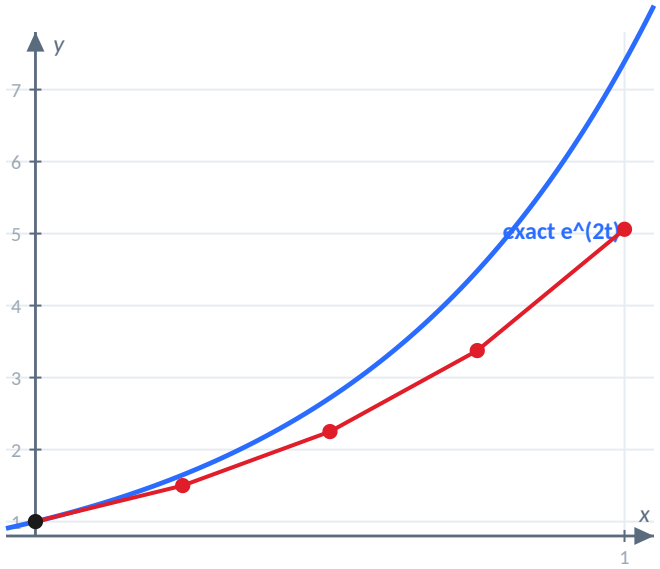
Example 5

Use one Euler step of size $\Delta t = 0.1$ to advance $\dfrac{dy}{dt} = 2y$ from the start $y_0 = 1$ at $t = 0$.

SOLUTION

1. At the start the rate is $\dfrac{dy}{dt} = 2y = 2 \times 1 = \boxed{2}$. current slope
2. Step formula: $y_{\text{next}} = y_{\text{now}} + \Delta t \cdot (\text{rate}) = 1 + 0.1 \times \boxed{2}$. new = old + dt·slope
3. Compute: $1 + 0.2 = \boxed{1.2}$.
4. Compare with the exact $e^{2\times 0.1} = e^{0.2} \approx 1.221$ – the single step is close, and smaller steps would be closer still.

After one step, $y \approx \boxed{1.2}$ (exact value ≈ 1.221).



Euler's method (red): short straight steps, each following the current slope, trace out the true exponential curve (blue). Smaller steps would hug it even tighter.

Euler's steps are little straight dashes that chase the true curve. They drift slightly because each dash assumes the slope holds steady for the whole step, when really it is changing. Shrink Δt and the dashes shorten, the drift shrinks, and the broken path melts into the smooth solution.

More worked examples

Example 6

A population obeys $\frac{dy}{dt} = k\,y$ and **doubles** every 5 hours. Find the growth constant k.

SOLUTION

1. Start from the solution $y(t) = y_0 e^{kt}$. "Doubles in 5 hours" means $y(5) = 2y_0$. translate the words
2. Substitute: $y_0 e^{5k} = 2y_0$. Cancel the y_0 on both sides: $e^{5k} = 2$. ÷ both sides by y0
3. Undo the exponential with a natural log: $5k = \ln 2$. ln both sides
4. Since $\ln 2 \approx 0.693$, divide by 5: $k = \frac{\ln 2}{5} \approx 0.139$.

The growth constant is $k = \frac{\ln 2}{5} \approx 0.139$ per hour – a tidy link between a doubling time and the rate that drives it.

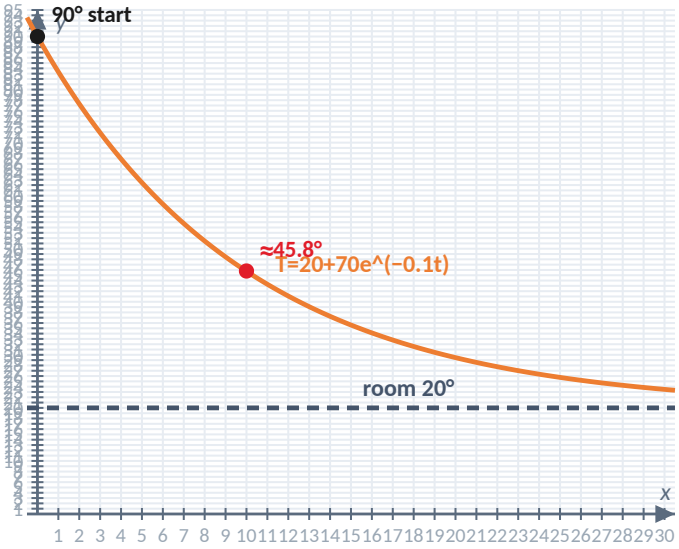
Example 7

Newton's law of cooling. A cup of coffee at 90° cools in a 20° room by $\frac{dT}{dt} = -0.1\,(T - 20)$. Show $T(t) = 20 + 70\,e^{-0.1t}$ fits, then find the temperature at $t = 10$.

SOLUTION

1. The room temperature 20 shifts the equation, so try the shifted form $T(t) = 20 + C\,e^{-0.1t}$. Differentiate: $\frac{dT}{dt} = -0.1\,C\,e^{-0.1t}$ (the constant 20 has slope 0). constant → 0
2. The right-hand side is $-0.1\,(T - 20) = -0.1\,(20 + Ce^{-0.1t} - 20) = -0.1\,C\,e^{-0.1t}$ – it matches, so the form solves the equation. both sides match
3. Fix C from the start $T(0) = 90$: $20 + Ce^0 = 20 + C = 90$, so $C = 70$, giving $T(t) = 20 + 70e^{-0.1t}$. use the start
4. At $t = 10$: $T = 20 + 70e^{-1}$. Since $e^{-1} \approx 0.368$, this is $20 + 70(0.368) = 20 + 25.8 = 45.8^\circ$.

The model fits, and the coffee is about 45.8° after 10 units of time – cooling fast at first, then easing as it nears the room's 20°.



Newton cooling from Example 7: the coffee falls steeply from 90° and levels off toward the room line at 20°, never quite reaching it – a shifted exponential decay.

Example 8

Take **two** Euler steps of size $\Delta t = 0.5$ to advance $\frac{dy}{dt} = y$ from $y_0 = 1$ at $t = 0$. Compare with the exact e^t at $t = 1$.

SOLUTION

1. Step 1. Current rate $= \frac{dy}{dt} = y =$ 1. New value $= 1 + 0.5 \times$ 1 $=$ 1.5; the clock moves to $t = 0.5$.

   new = old + dt·slope

2. Step 2. Now the rate uses the *updated* y: rate $= y =$ 1.5. New value $= 1.5 + 0.5 \times$ 1.5 $= 1.5 +$ $0.75 =$ 2.25; the clock reaches $t = 1$. *re-read the slope*
3. The exact answer is $e^1 \approx 2.718$, so two coarse steps give 2.25 – in the right region but low, because each step under-uses the rising slope.
4. Halving the step (four steps) would climb noticeably closer to 2.718.

After two steps, $y \approx$ 2.25 versus the true $e \approx 2.718$ – coarse steps under-shoot, finer steps close the gap.

Mental-Math Shortcut

Spot the equation $\frac{dy}{dt} = ky$ and you can skip straight to the answer: $y = y_0 e^{kt}$. Read the **sign of** k to know the story – plus means **grow**, minus means **decay** – and read its **size** for the drama: a large $|k|$ means a steep curve, a small one a gentle drift. No calculation needed to know the shape.

Why is the solution of $\frac{dy}{dt} = ky$ *forced* to be exponential, and not merely "one option among many"? The reasoning is a tidy uniqueness argument.

First, existence: we already checked that $y = y_0 e^{kt}$ satisfies the rule, since differentiating it returns k times itself. So a solution *exists*. The real question is whether any **other** function could also work. Suppose a rival solution $g(t)$ also obeyed $g' = kg$ with the same start $g(0) = y_0$. Build the ratio $r(t) = \dfrac{g(t)}{y_0 e^{kt}}$ and differentiate it with the quotient rule:

$$r'(t) = \frac{g'\,(y_0 e^{kt}) - g\,(k\,y_0 e^{kt})}{(y_0 e^{kt})^2} = \frac{(kg)\,(y_0 e^{kt}) - g\,(k\,y_0 e^{kt})}{(y_0 e^{kt})^2} = 0.$$

Now reason it through. A function whose derivative is 0 *everywhere* cannot change – it is a flat constant. And at $t = 0$ that constant is $r(0) = \dfrac{y_0}{y_0} = 1$. So $r(t) = 1$ for all t, which means $g(t) = y_0 e^{kt}$. *Therefore* the rival was never a rival at all – it is the very same function. The rule plus the starting value pin down **exactly one** solution. That "if a derivative is zero everywhere then nothing changes" step is the logical engine that makes differential equations *predict* rather than merely *describe*.

⚠ Common Mistakes

- Expecting a number for the answer. The solution of a differential equation is a **function**, $y(t)$, not a single value.
- Dropping the starting amount y_0. The rule $\dfrac{dy}{dt} = ky$ alone is not enough; you also need where the function *starts* to fix it.
- Muddling the sign of k. Positive k is growth, negative k is decay – the sign sets the whole direction.
- Believing one big Euler step is exact. It is an **approximation**; accuracy improves only as the step Δt shrinks.
- Forgetting the chain rule when checking solutions: $\dfrac{d}{dt} e^{kt} = k\,e^{kt}$, not e^{kt}.

Quiz

60.1.Q1. What kind of object is the solution to a differential equation?

A. A function

B. A single number

C. A vector

D. A matrix

60.1.Q2. In words, what does $\frac{dy}{dt} = ky$ say?

60.1.Q3. What function solves $\frac{dy}{dt} = ky$ with start y_0?

A. $y_0 e^{kt}$

B. $ky + t$

C. $y_0 + kt$

D. kt^2

60.1.Q4. If k is negative, does y grow or decay over time?

60.1.Q5. A population follows $\frac{dy}{dt} = 0.2y$ from $y_0 = 50$. Write its solution.

60.1.Q6. Does $y = e^{3t}$ satisfy $\frac{dy}{dt} = 3y$?

A. Yes

B. No

60.1.Q7. In Euler's method, how do you get the next value from the current one?

60.1.Q8. One Euler step of size 0.1 on $\frac{dy}{dt} = 2y$ from $y = 1$ gives what?

A. 1.2

B. 2

C. 1.02

D. 0.2

60.1.Q9. Why does a smaller Euler step give a more accurate answer?

🐍 From Concept to Code (Python)

Euler's method is a loop made for computers. Start at y_0, and on each pass measure the current rate, nudge y by step·rate, and advance the clock. Below we march $\frac{dy}{dt} = 2y$ forward and compare with the exact e^{2t}.

```
import math

def euler(rate, y0, t_end, dt=0.001):
    y, t = y0, 0.0
    while t < t_end - 1e-9:
        y = y + dt * rate(t, y)      # step: new = old + dt * slope
        t = t + dt
    return y

rate = lambda t, y: 2 * y            # the rule dy/dt = 2y
approx = euler(rate, 1.0, 1.0)       # march from y=1 to t=1
exact  = math.exp(2 * 1.0)           # the true answer, e^2

print(round(approx, 4))
print(round(exact, 4))
```

```
output
7.3878
7.3891
```

Try it:

1. Make the step coarse, `dt=0.5`, and watch the approximation fall noticeably short of $e^2 \approx 7.389$.
2. Switch to decay with `rate = lambda t, y: -0.5 * y` and confirm the value at $t = 2$ is near $e^{-1} \approx 0.368$.
3. Have the loop also `print` each `(t, y)` so you can see the little straight steps climbing the curve.

Answer Key – Chapter 60 – Differential Equations

60.1.Q1. A function – e.g. $y(t)$ – whose rate of change obeys the equation.

60.1.Q2. The rate of change of y is proportional to y itself – the bigger y is, the faster it changes.

60.1.Q3. $y(t) = y_0 e^{kt}$ – exponential growth or decay.

60.1.Q4. It decays – fading toward zero.

60.1.Q5. $y(t) = 50e^{0.2t}$.

60.1.Q6. Yes – differentiating gives $3e^{3t} = 3y$, matching the equation.

60.1.Q7. Add the time step times the current rate: $y_{\text{next}} = y_{\text{now}} + \Delta t \cdot \frac{dy}{dt}$.

60.1.Q8. 1.2 – rate is $2 \times 1 = 2$, so $1 + 0.1 \times 2 = 1.2$.

60.1.Q9. Each step assumes the slope is constant over it; shorter steps make that assumption truer, so error shrinks.

Chapter 61 · Series & Approximations

How adding up endlessly many shrinking pieces can land on a single finite total, and how stacking a few simple powers of x can imitate a complicated curve.

The Idea Behind It

Adding up *infinitely* many shrinking powers lands on a clean finite total: $1 + r + r^2 + r^3 + \cdots = \dfrac{1}{1-r}$ **for** $|r| < 1$. *but why?*

Give the unknown total a name and pin it down with one clever move: write the whole sum, then write a *shifted copy* of it, and subtract so that almost everything cancels. The fact we lean on is that a shifted copy is just the sum multiplied by r:

$$\boxed{S = 1 + r + r^2 + r^3 + \cdots \quad \Rightarrow \quad rS = r + r^2 + r^3 + \cdots}$$

Look at rS carefully: it is the original S with the leading 1 chopped off, so $rS = S - 1$. Subtract it from S and every matching power telescopes away, leaving only the 1:

$$S - rS = (1 + r + r^2 + \cdots) - (r + r^2 + \cdots) = 1$$

Factor the left side and solve the now-simple equation for S:

$$S\,(1 - r) = 1 \quad \Longrightarrow \quad S = \frac{1}{1-r}$$

$$\therefore \; 1 + r + r^2 + r^3 + \cdots = \frac{1}{1-r}$$

The condition $|r| < 1$ is what makes the cancellation honest: only then do the powers r^n shrink toward 0, so the leftover tail vanishes and the sum truly **converges**. If $|r| \geq 1$ the terms do not fade and the total runs off to infinity. *subtract a shifted copy; the middle telescopes away*

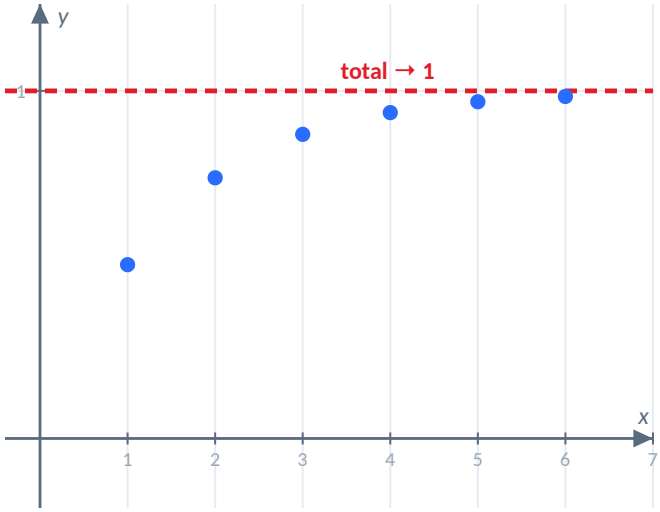
This final chapter ties calculus to a surprising idea: you can add up *infinitely many* numbers and still get a sensible, finite answer – and you can build a complicated function out of nothing but simple powers of x. These two ideas, **infinite series** and **polynomial approximation**, are how a calculator finds e, how it computes a sine, and how a great deal of scientific computing actually works under the hood.

Adding forever, yet stopping somewhere

Start with one whole pizza. Eat half. Then half of what is left, then half of *that*, on and on forever. How much do you eat in total? The slices are $\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \cdots$, an endless list. Common sense says you can never eat more than the one pizza – and the total is exactly 1. An infinite sum can have a finite value.

INFINITE SERIES

The sum of an unending list of terms, $a_1 + a_2 + a_3 + \cdots$. If the running total settles toward one fixed number as you keep adding, the series **converges** to that number; if the total runs off forever, it **diverges**.



Running totals of ½+¼+⅛+... (blue dots) climb toward the red line at 1 but never cross it. The series converges to 1.

That pizza sum is a **geometric series**: each term is a fixed fraction of the one before it. Here we multiply by $r = \frac{1}{2}$ every step. Whenever that ratio satisfies $|r| < 1$, the terms shrink fast enough for the whole infinite sum to land on a clean value:

$$a + ar + ar^2 + ar^3 + \cdots = \frac{a}{1 - r} \quad (\text{valid when } |r| < 1)$$

Check the pizza: first term $a = \frac{1}{2}$, ratio $r = \frac{1}{2}$, so the sum is $\frac{1/2}{1 - 1/2} = \frac{1/2}{1/2} = 1$. The formula confirms the picture.

Note: The condition $|r| < 1$ is essential. If each term is as big as or bigger than the last (say $r = 2$: $1 + 2 + 4 + 8 + \cdots$), the total grows without bound – the series diverges and the formula does not apply.

Worked examples

Example 1

Sum the infinite series $\frac{1}{3} + \frac{1}{9} + \frac{1}{27} + \cdots$.

SOLUTION

1. Identify it as geometric: each term is $\frac{1}{3}$ of the one before, so the ratio is $r = \frac{1}{3}$. |r|<1 → converges
2. The first term is $a = \frac{1}{3}$. Since $|r| < 1$, the sum formula applies.
3. Compute $\frac{a}{1 - r} = \frac{1/3}{1 - 1/3} = \frac{1/3}{2/3}$. a ÷ (1−r)
4. Dividing fractions: $\frac{1/3}{2/3} = \frac{1}{2}$.

The sum is $\frac{1}{2}$.

Example 2

Sum $5 + \frac{5}{2} + \frac{5}{4} + \frac{5}{8} + \cdots$.

SOLUTION

1. Each term is half the previous, so $r = \frac{1}{2}$, and the first term is $a = 5$. read off a and r
2. Apply $\frac{a}{1-r} = \frac{5}{1-1/2} = \frac{5}{1/2}$. a ÷ (1–r)
3. Dividing by $\frac{1}{2}$ doubles: $5 \times 2 = 10$.

The sum is 10.

Example 3

Does $1 + 2 + 4 + 8 + \cdots$ converge?

SOLUTION

1. Each term is twice the one before, so $r = 2$.
2. The test demands $|r| < 1$, but here $|r| = 2$. |r|≥1 → diverges
3. The terms grow instead of shrinking, so the running total races off to infinity.

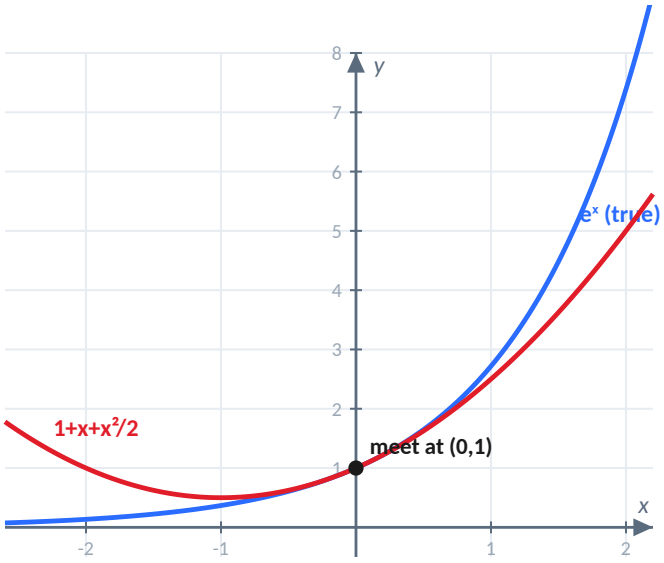
No – it diverges; there is no finite sum.

Building a curve out of powers of x

Here is the deeper idea. Many functions are "curvy" and awkward, but a **polynomial** – just a sum of powers of x like $1 + x + x^2$ – is easy: a calculator only needs to multiply and add. So we *approximate* a hard function with a polynomial that matches it closely near a chosen point. The exponential e^x has a famously tidy one:

$$e^x \approx 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \cdots$$

Each extra term sharpens the fit. Keep just the first three, $1 + x + \frac{x^2}{2}$, and you get a simple parabola that **hugs** e^x tightly near $x = 0$ and drifts away only as you move further out.



The red parabola 1+x+x²/2 clings to the blue curve e^x near x=0 and only peels away far from the origin. More terms would extend the hug.

POLYNOMIAL APPROXIMATION

Replacing a function near a point by a polynomial whose value, slope, curvature and so on agree with the function there. Near that point the two are nearly indistinguishable; the more matching terms, the wider the agreement.

Example 4

Estimate $e^{0.1}$ using the three-term approximation $1 + x + \frac{x^2}{2}$.

SOLUTION

1. Substitute $x = 0.1$ into the polynomial. *substitute x*
2. $1 + 0.1 + \dfrac{0.1^2}{2} = 1 + 0.1 + \dfrac{0.01}{2}$.
3. $= 1 + 0.1 + 0.005 = 1.105$.
4. A calculator gives $e^{0.1} \approx 1.10517$ – the three terms are accurate to four decimals because 0.1 is close to 0.

$e^{0.1} \approx 1.105$ **, matching the true value almost exactly.**

Example 5

Use the same polynomial to estimate e^1, then say why it is less accurate.

SOLUTION

1. Substitute $x = 1$: $1 + 1 + \frac{1^2}{2} = 1 + 1 + 0.5 = 2.5$. *substitute x*
2. The true value is $e \approx 2.718$.
3. The estimate 2.5 is in the right area but noticeably low, because $x = 1$ is *far* from the point $x = 0$ where the polynomial was tuned. *far from centre*

About 2.5 versus the true 2.718 – decent, but the approximation weakens as x moves away from 0.

Example 6

Add the next term, $\frac{x^3}{6}$, and re-estimate e^1.

SOLUTION

1. Now use $1 + x + \frac{x^2}{2} + \frac{x^3}{6}$ at $x = 1$. *one more term*
2. $1 + 1 + 0.5 + \frac{1}{6} = 2.5 + 0.1667$.
3. $= 2.6667$.
4. Closer to 2.718 than the three-term answer – each extra power tightens the fit.

≈ 2.667 **– one more term, one step nearer the truth.**

Example 7

Write the repeating decimal $0.7777\ldots$ as an exact fraction using a geometric series.

SOLUTION

1. Split the decimal into its digit-pieces: $0.7 + 0.07 + 0.007 + \cdots$, a sum of shrinking sevenths-of-tens. *unpack the decimal*
2. This is geometric: the first term is $a = 0.7$ and each step multiplies by $r = 0.1$. Since $|r| < 1$, the sum exists.
3. Apply $\frac{a}{1 - r} = \frac{0.7}{1 - 0.1} = \frac{0.7}{0.9}$. *a ÷ (1–r)*
4. Clear the decimals by multiplying top and bottom by 10: $\frac{7}{9} = \frac{7}{9}$.

$0.7777\ldots = \frac{7}{9}$ **exactly – every repeating decimal is secretly a geometric series.**

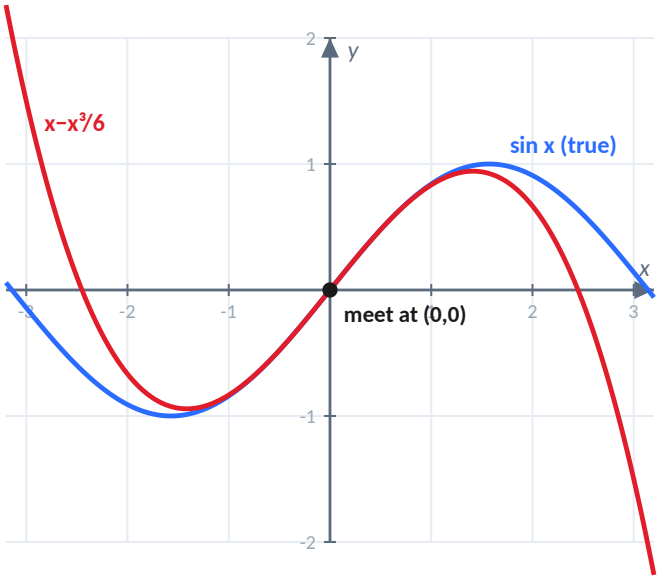
Example 8

The sine function has the polynomial $\sin x \approx x - \frac{x^3}{6}$ near $x = 0$. Estimate $\sin(0.5)$ (radians) and compare.

SOLUTION

1. Substitute $x = 0.5$ into the polynomial. substitute x
2. First term $x = 0.5$; second term $\frac{x^3}{6} = \frac{0.5^3}{6} = \frac{0.125}{6} = 0.0208$.
3. Subtract: $0.5 - 0.0208 = 0.4792$. $x - x^3/6$
4. A calculator gives $\sin(0.5) \approx 0.4794$ – just two terms already match to three decimals, because 0.5 is close to 0.

$\sin(0.5) \approx 0.479$, **all but identical to the true** 0.4794 **– the same hug-near-zero idea, now for sine.**



The red polynomial $x - x^3/6$ traces sine (blue) almost perfectly near x=0 and only peels away past about ±1.5 (Example 8). More terms would lengthen the match.

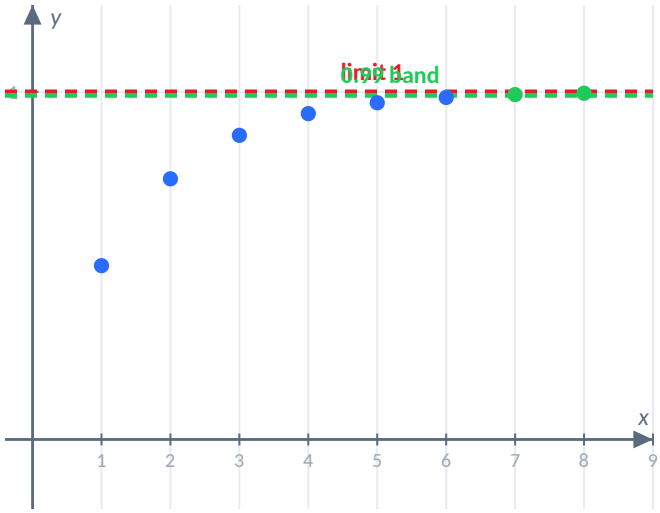
Example 9

How many terms of the pizza series $\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \cdots$ are needed before the running total is within 0.01 of its limit 1?

SOLUTION

1. The sum of the first n terms is $S_n = 1 - \dfrac{1}{2^n}$ (each step closes half the remaining gap), so the leftover gap to 1 is exactly $\dfrac{1}{2^n}$. gap = 1 − Sn
2. We need that gap under 0.01: $\dfrac{1}{2^n} < 0.01$, i.e. $2^n > 100$. set the gap < 0.01
3. Count powers of two: $2^6 = 64$ (still under 100) but $2^7 = 128 > 100$. compare to 100
4. So $n =$ 7 terms first clear the bar; then $S_7 = 1 - \dfrac{1}{128} \approx$ 0.992.

After 7 terms the total reaches about 0.992, within 0.01 of the limit 1 – the convergence dots below cross into that band at the seventh point.



Running totals from Example 9. The seventh dot (green) is the first to rise above the 0.99 band, so seven terms land within 0.01 of the limit 1.

Tip: More terms widen and tighten the hug; staying near the centre point keeps even a short polynomial extremely accurate. This trade-off – a few terms near 0 – is exactly how calculators and libraries compute e^x, $\sin x$ and friends.

Series and approximations close the loop of the whole subject. A derivative measured an instant, an integral summed a region, a differential equation predicted the future – and here, an infinite series lets a machine compute the very numbers those tools call for, using only the four basic operations of arithmetic you began with.

⚡ Mental-Math Shortcut

For any geometric series with first term a and ratio r, the infinite sum is simply $\frac{a}{1-r}$ – *provided* $|r| < 1$. A quick instinct: if the terms are visibly **shrinking**, a finite total exists; if they hold steady or **grow**, it blows up. And for a polynomial approximation, accuracy is best **closest to the centre point** ($x = 0$ for e^x) – the nearer you stay, the fewer terms you need.

Why should an *endless* addition equal a single number, rather than "infinity" or nonsense? The geometric series gives an airtight argument using a clever cancellation.

Add only the first n terms and call that partial total $S_n = a + ar + ar^2 + \cdots + ar^{n-1}$. Multiply the whole line by r:

$$rS_n = ar + ar^2 + \cdots + ar^{n-1} + ar^n.$$

Subtract the second line from the first. Almost everything cancels in the middle – every shared term disappears – leaving just the two ends:

$$S_n - rS_n = a - ar^n \quad \Longrightarrow \quad S_n = \frac{a\,(1 - r^n)}{1 - r}.$$

Now reason it through. With $|r| < 1$, raising r to higher and higher powers drives r^n toward 0 – multiply a number smaller than 1 by itself enough times and it withers away. *Therefore* the r^n term vanishes in the limit, and S_n is squeezed toward $\frac{a\,(1-0)}{1-r} = \frac{a}{1-r}$. The infinite sum is **defined** as that limit of partial totals. The "watch what the partial sums approach" move is Chapter 54's limit one last time – the same idea that opened calculus now closes it.

⚠ Common Mistakes

- Assuming every infinite sum is infinite. When the terms shrink fast enough ($|r| < 1$), the total is a finite number.
- Using $\frac{a}{1-r}$ when $|r| \geq 1$. The formula only holds for a **shrinking** geometric series; otherwise it diverges.
- Forgetting that a is the **first** term you are actually adding. In $5 + \frac{5}{2} + \cdots$ the first term is 5, not 1.
- Expecting a short polynomial to match a function *everywhere*. The approximation is excellent **near the centre point** and worsens as you move away.
- Stopping too early when high accuracy is needed. Each extra term of $1 + x + \frac{x^2}{2} + \cdots$ tightens the fit; far from 0 you need more of them.

Quiz

61.1.Q1. What does it mean for an infinite series to "converge"?

61.1.Q2. Sum the geometric series $\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \cdots$.

A. 1

B. 2

C. $\frac{1}{2}$

D. infinite

61.1.Q3. For what ratios r does $\dfrac{a}{1-r}$ give the sum of a geometric series?

61.1.Q4. Find the sum of $\frac{1}{3} + \frac{1}{9} + \frac{1}{27} + \cdots$.

61.1.Q5. Does $1 + 3 + 9 + 27 + \cdots$ converge?

A. No, it diverges

B. Yes, to a finite value

61.1.Q6. Estimate e^x near $x = 0$ with three terms.

A. $1 + x + \frac{x^2}{2}$

B. $1 + x^2 + x^3$

C. $x + x^2$

D. $1 - x + x^2$

61.1.Q7. Use $1 + x + \frac{x^2}{2}$ to approximate $e^{0.2}$.

61.1.Q8. Why is $1 + x + \frac{x^2}{2}$ a worse estimate of e^x at $x = 2$ than at $x = 0.1$?

61.1.Q9. True or False: adding more terms to a polynomial approximation generally improves the fit.

A. True

B. False

From Concept to Code (Python)

Two short programs capture the chapter. The first **sums a geometric series** term by term and watches it approach $\frac{a}{1-r}$. The second **builds the polynomial** for e^x by adding more and more powers, closing in on Python's own value of e^x.

```
def geometric_sum(a, r, terms):
    total, term = 0.0, a
    for _ in range(terms):
        total += term               # add this term
        term *= r                   # next term is r times this one
    return total

# pizza series: a = 1/2, r = 1/2, should approach 1
print(round(geometric_sum(0.5, 0.5, 5),  4))
print(round(geometric_sum(0.5, 0.5, 20), 4))
```

```
output
0.9688
1.0
```

```
import math

def exp_approx(x, terms):
    total, term = 0.0, 1.0          # first term of eˣ is 1
    for k in range(terms):
        total += term               # add x^k / k!
        term *= x / (k + 1)         # build the next term cheaply
    return total

print(round(exp_approx(1.0, 3), 4))   # three terms
print(round(exp_approx(1.0, 8), 4))   # eight terms
print(round(math.exp(1.0),    4))     # Python's true e¹
```

```
output
2.5
2.7183
2.7183
```

Try it:

1. Run `geometric_sum(5, 0.5, 30)` and confirm it approaches 10 (Example 2).
2. Try `geometric_sum(1, 2, 10)` – with $r = 2$ the total explodes, showing divergence.
3. Call `exp_approx(0.1, 3)` and compare with `math.exp(0.1)`: near 0, three terms already nail it.

Answer Key – Chapter 61 – Series & Approximations

61.1.Q1. Its running total settles toward one fixed, finite number as more terms are added.

61.1.Q2. 1 – with $a = \frac{1}{2}$, $r = \frac{1}{2}$, the sum is $\dfrac{1/2}{1 - 1/2} = 1$.

61.1.Q3. When $|r| < 1$ – the terms must shrink for the total to be finite.

61.1.Q4. $\dfrac{1}{2}$ – first term $\frac{1}{3}$, ratio $\frac{1}{3}$, so $\dfrac{1/3}{2/3} = \frac{1}{2}$.

61.1.Q5. No – the ratio is 3, so terms grow and the total runs to infinity.

61.1.Q6. $1 + x + \dfrac{x^2}{2}$ – the start of the exponential's polynomial.

61.1.Q7. 1.22 – $1 + 0.2 + \dfrac{0.04}{2} = 1.22$, very close to the true 1.2214.

61.1.Q8. The polynomial is tuned to match near $x = 0$; accuracy fades the further x is from that centre.

61.1.Q9. True – each extra power makes it hug the true curve over a wider range.



PART XI

Number Bases & the Language of Computers

How counting itself depends on a chosen base. Why humans settled on ten, how every base reuses a handful of symbols through place value, and how computers run the entire digital world on just two symbols: 0 and 1.

Chapter 62 – Number Bases: A Primer · Chapter 63 – Base 10: The Decimal System · Chapter 64 – Base 2: Binary & How Computers Count

Chapter 62 · Number Bases: A Primer

> *What a number base really is: how a handful of digits plus the idea of place value lets us write every number, how to read and convert between bases (binary, decimal, hexadecimal), and why the choice of base is a human convenience, not a law of nature.*

The Idea Behind It

The same digits "101" mean five in binary but one hundred and one in decimal. *but why?*

A *base* tells you what each column is worth. In base b the columns going right to left are the powers $b^0, b^1, b^2, \ldots$, and a string of digits is shorthand for adding each digit times its column. That one rule, **positional value**, defines every base at once:

$$\boxed{d_k \ldots d_2 d_1 d_0 \;=\; d_k b^k + \cdots + d_2 b^2 + d_1 b^1 + d_0 b^0}$$

So read "101" in two systems just by choosing b. In **binary** ($b = 2$) the columns are $4, 2, 1$; in **decimal** ($b = 10$) they are $100, 10, 1$:

$$101_2 = 1\cdot 2^2 + 0\cdot 2^1 + 1\cdot 2^0 = 4 + 0 + 1 = 5$$

$$101_{10} = 1\cdot 10^2 + 0\cdot 10^1 + 1\cdot 10^0 = 100 + 0 + 1 = 101$$

$\therefore$ digits alone are not a number, the **base** is half the message. Same symbols, different column values, different number. *the base sets the column values*

Count on your fingers and you reach 10 before you run out of symbols. That is no accident: our everyday system is **base ten**, built from exactly ten digits $0, 1, 2, \ldots, 9$. But ten is a *choice*, not a necessity. Computers count in **base two**, programmers often read **base sixteen**, and old clocks still run on **base sixty**. This chapter unpacks the single idea that makes all of them work: **place value**.

BASE (RADIX)

The **base** (or **radix**) of a number system is how many distinct digits it uses, and therefore the value of each successive place. Base b uses the digits 0 up to $b - 1$, and reading right to left the columns are worth $b^0, b^1, b^2, \ldots$. We tag a number with its base as a subscript: 1011_2, 347_{10}, $2\mathrm{A}_{16}$.

Why we need a base at all

We have only a handful of symbols but want to name endless numbers. The trick is to **reuse** the symbols in columns, each column worth the base times the one to its right. When a column fills up it **carries** into the next, exactly like a car odometer rolling $009 \to 010$. The base is simply *when* each wheel rolls over.

Reading any base: expand by place value

To find what a string of digits is worth, multiply each digit by its column ($b^0, b^1, \ldots$) and add. This always lands you back in familiar base ten.

$$1011_2 = 1\cdot 2^3 + 0\cdot 2^2 + 1\cdot 2^1 + 1\cdot 2^0 = 8 + 0 + 2 + 1 = 11$$

The common bases

Base	Name	Digits used	Where you meet it
2	Binary	0, 1	Computers, digital logic
8	Octal	0 to 7	Older computing, file permissions
10	Decimal	0 to 9	Everyday human counting
16	Hexadecimal	0 to 9, A to F	Colours, memory addresses

Note: In hexadecimal the single digits run out at 9, so we borrow letters: $\mathrm{A} = 10$, $\mathrm{B} = 11$, $\mathrm{C} = 12$, $\mathrm{D} = 13$, $\mathrm{E} = 14$, $\mathrm{F} = 15$. One hex digit packs the values 0 to 15.

Converting base ten into another base: repeated division

Going the other way, **divide by the base repeatedly and collect the remainders**. Each remainder is one digit, because it is exactly what would not fit into the next-higher column. Read the remainders **bottom to top**.

Worked examples

Example 1

Convert 1011_2 to decimal.

SOLUTION

1. Write the column values (powers of 2): $8\ 4\ 2\ 1$.
2. Keep the columns that hold a 1: $8 + 0 + 2 + 1 = 8 + 2 + 1$.

$1011_2 = 11$

Example 2

Convert 13_{10} to binary by repeated division by 2.

SOLUTION

1. $13 \div 2 = 6$ r 1, then $6 \div 2 = 3$ r 0, then $3 \div 2 = 1$ r 1, then $1 \div 2 = 0$ r 1.
2. Read the remainders **bottom to top**: 1101.

$13_{10} = 1101_2$ **(check:** $8 + 4 + 1 = 13$**)**

Example 3

Convert 2A_{16} to decimal.

SOLUTION

1. Hex columns are 16^1 and 16^0, and the digit A is 10.
2. $2 \cdot 16 + 10 \cdot 1 = 32 + 10$.

$2\text{A}_{16} = 42$

Example 4

Convert 255_{10} to hexadecimal.

SOLUTION

1. $255 \div 16 = 15$ r 15, then $15 \div 16 = 0$ r 15.
2. Both remainders are $15 = \text{F}$; read bottom to top: FF.

$255_{10} = \text{FF}_{16}$ **(the largest two-digit hex number)**

Example 5

In base 5 (digits 0 to 4), what does 24_5 mean in decimal?

SOLUTION

1. Columns are 5^1 and 5^0: $2 \cdot 5 + 4 \cdot 1 = 10 + 4$.

$24_5 = 14$

Example 6

Why is "19_2" impossible to write?

SOLUTION

1. Base 2 only owns the digits 0 and 1.
2. A "9" can never appear in binary, just as a "G" cannot appear in hexadecimal.

No digit may be as large as its base: in base b the digits stop at $b - 1$.

Example 7

Count from zero in base 2 up to decimal 5.

SOLUTION

1. Each step adds 1; when a column reaches 2 it rolls over and carries.
2. Zero, one, then $1+1$ carries to 10, then 11, then 100, then 101.

$0, 1, 10, 11, 100, 101$ $=$ **decimal** $0, 1, 2, 3, 4, 5$

Note: A subscript is essential whenever confusion is possible. "10" is **ten** in decimal, **two** in binary, **sixteen** in hex, and in fact equals the base in *every* system, because $10_b = 1 \cdot b + 0 = b$.

base radix digit place value hexadecimal

⚡ Mental-Math Shortcut

Spot a base from its digits. If you see a 7, it is not binary. If you see a letter such as C, it is hexadecimal. Binary is all 0s and 1s; decimal stops at 9.

Powers to know cold. Powers of two: $1, 2, 4, 8, 16, 32, 64, 128, 256, \ldots$ Powers of sixteen begin $1, 16, 256, 4096$. Knowing these lets you expand short numbers in your head.

The "10 equals the base" trick. In any base, 10 is the base itself, 100 is the base squared, and so on. So $1000_2 = 2^3 = 8$ and $1000_{16} = 16^3 = 4096$ with no arithmetic.

🧠 The Logic of It

Counting is an *algorithm*, and the base is its only setting. Start at 0; to add one, bump the rightmost digit; if it would reach the base, set it to 0 and **carry** a bump into the next column, repeating leftwards. That single rule generates decimal, binary, and every other system, the way a row of odometer wheels of different sizes would. Changing the base does not change the *logic* of counting, only the size of each wheel.

⚠ Common Mistakes

- Reading digits without their base. "101" is meaningless until you know the base: it is 5 in binary but 101 in decimal.
- Using a digit equal to or larger than the base, such as 1012_2 or G_{16}. Base b stops at the digit $b - 1$.
- Reading repeated-division remainders top to bottom. The **first** remainder is the **last** (rightmost) digit, so read them bottom to top.
- Forgetting that $b^0 = 1$, so the rightmost column is always the **ones** column in every base.
- Treating hexadecimal letters as variables. A to F are fixed digits worth 10 to 15.

Quiz

62.1.Q1. What is 110_2 in decimal?

A. 4

B. 6

C. 110

D. 3

62.1.Q2. Convert 20_{10} to binary.

62.1.Q3. What is the largest digit available in base 8?

A. 8

B. 7

C. 9

D. F

62.1.Q4. Convert $1\mathrm{F}_{16}$ to decimal.

62.1.Q5. In base 3, what is 21_3 in decimal?

62.1.Q6. True or False: "10" equals the base in every number system.

A. True

B. False

62.1.Q7. Which of these cannot be a binary number: 1001, 1021, 1110?

62.1.Q8. Convert 100_{10} to hexadecimal.

62.1.Q9. How many distinct digits does base 16 use?

A. 15

B. 16

C. 10

D. 6

🐍 From Concept to Code (Python)

Python speaks every common base. Literals can carry a prefix, `0b` for binary, `0o` for octal, `0x` for hex, and the built-ins `bin()`, `oct()`, `hex()` convert a number *to* a base-tagged string, while `int(text, base)` reads a string *from* any base back into an ordinary integer.

```
print(0b1011)          # a binary literal
print(0x2A)            # a hex literal
print(int("1101", 2))  # read base-2 text
print(int("FF", 16))   # read base-16 text
```

```
output
11
42
13
255
```

```
n = 13
print(bin(n))    # to binary
print(hex(255))  # to hex
print(oct(64))   # to octal
```

```
output
0b1101
0xff
0o100
```

Try it:

1. Predict, then run: `print(int("10", 2), int("10", 8), int("10", 16))`. Why are they all different?
2. Use `int(input(), 2)` to read a binary string from the user and print its decimal value.

Answer Key – Chapter 62 – Number Bases: A Primer

62.1.Q1. 6 – columns $4 + 2 + 0$.

62.1.Q2. 10100_2 – it is $16 + 4$ (remainders of repeated division by 2).

62.1.Q3. 7 – base b stops at $b - 1$.

62.1.Q4. 31 – $1 \cdot 16 + 15$.

62.1.Q5. 7 – $2 \cdot 3 + 1$.

62.1.Q6. True – $10_b = 1 \cdot b + 0 = b$.

62.1.Q7. 1021 – it contains a 2, which binary does not have.

62.1.Q8. 64_{16} – $6 \cdot 16 + 4 = 100$.

62.1.Q9. 16 – the digits 0 to 9 and A to F.

Chapter 63 · Base 10: The Decimal System

> *Why humans count in tens, how decimal place value is built from powers of ten, and why base ten is so convenient across science, engineering and mathematics, from the metric system to scientific notation and orders of magnitude.*

The Idea Behind It

Each step left in a decimal number multiplies its worth by ten, so $347 = 300 + 40 + 7$. *but why?*

Decimal is base ten, so its column values are the **powers of ten**: $1, 10, 100, 1000, \ldots$ Reading a number is just adding each digit times its column, which is positional value with $b = 10$:

$$\boxed{d_k \ldots d_1 d_0 = d_k 10^k + \cdots + d_1 10^1 + d_0 10^0}$$

So the 3 in 347 is not really "three", it is **three hundreds**, because it sits in the 10^2 column:

$$347 = \boxed{3} \cdot 10^2 + 4 \cdot 10^1 + 7 \cdot 10^0 = 300 + 40 + 7$$

$\therefore$ the *position* carries as much meaning as the digit itself. That is why a single zero matters so much: 40 and 400 differ only by shoving the digits into higher power-of-ten columns. *position = a power of ten*

Why ten? Look down at your hands. Almost every human culture that counted on fingers arrived at groups of **ten**, and our word *digit* means both "number-symbol" and "finger". Base ten is an accident of anatomy, not a mathematical truth, yet it has shaped science so deeply that the whole metric system is built to match it.

DECIMAL (BASE TEN)

The **decimal** system is base 10: ten digits 0 to 9, with each column worth ten times the one to its right. Going left from the **decimal point** the columns are ones, tens, hundreds ($10^0, 10^1, 10^2, \ldots$); going right they are tenths, hundredths ($10^{-1}, 10^{-2}, \ldots$).

Place value is powers of ten

Every decimal number is a sum of digits weighted by powers of ten. This is exactly why "adding a zero" multiplies a whole number by ten: it shifts every digit one column to the left, into the next power.

$$2{,}506 = 2 \cdot 10^3 + 5 \cdot 10^2 + 0 \cdot 10^1 + 6 \cdot 10^0$$

A short history of ten (and its rivals)

Ten is not the only base humans have used. The Babylonians counted in **sixty**, which is why an hour still has 60 minutes and a circle 360 degrees. The Maya used **twenty** (fingers *and* toes). Even English keeps fossils of other bases: a *dozen* is twelve, a *score* is twenty. Ten simply won because of our fingers.

Why base ten is a gift to STEM

Because decimal columns are powers of ten, anything else built on powers of ten meshes with it perfectly: - **The metric (SI) system.** Prefixes are pure powers of ten, so converting is just sliding the decimal point: $1\ \text{km} = 10^3\ \text{m}$

and $1\ \text{mm} = 10^{-3}\ \text{m}$. - **Scientific notation.** Any quantity becomes (a number between 1 and 10) $\times\ 10^{n}$, so a galaxy ($10^{21}\ \text{m}$ across) and an atom ($10^{-10}\ \text{m}$) fit on the same page. - **Orders of magnitude.** Each extra digit is one more power of ten, so you can compare sizes at a glance: a quantity with 7 digits is "millions", roughly 10^{6}.

> **Tip:** The number of digits in a whole number reveals its order of magnitude. A 4-digit number is about 10^{3} (thousands); a 10-digit number is about 10^{9} (billions).

Worked examples

Example 1

Expand $4{,}073$ by place value.

SOLUTION

1. Tag each digit with its power of ten: thousands, hundreds, tens, ones.
2. $4 \cdot 10^{3} + 0 \cdot 10^{2} + 7 \cdot 10^{1} + 3 \cdot 10^{0}$.

$4000 + 0 + 70 + 3 = 4073$

Example 2

Convert 2.5 kilometres to metres using powers of ten.

SOLUTION

1. "Kilo" means 10^{3}, so $1\ \text{km} = 1000\ \text{m}$.
2. Multiply: 2.5×10^{3}, which slides the point three places right.

$2.5\ \text{km} = 2500\ \text{m}$

Example 3

Write $93{,}000{,}000$ (miles to the Sun) in scientific notation.

SOLUTION

1. Put the point just after the first non-zero digit: 9.3.
2. Count how far it moved: 7 places left, so multiply by 10^{7}.

9.3×10^{7} **miles**

Example 4

Which is larger, 10^6 or $9{,}000{,}000$, and by roughly how many times?

SOLUTION

1. $10^6 = 1{,}000{,}000$ (seven digits), and $9{,}000{,}000$ also has seven digits.
2. So $9{,}000{,}000 = \boxed{9} \times 10^6$, which is 9 copies of 10^6.

$9{,}000{,}000$ **is larger, about** $\boxed{9}$ **times** 10^6**.**

Example 5

How many tens make one hundred, and why?

SOLUTION

1. The hundreds column is worth $10^2 = 100$ and the tens column is $10^1 = 10$.
2. Each step up the columns multiplies by $\boxed{10}$, so one hundred is ten tens.

$\boxed{10}$ **tens make one hundred (each column is ten of the next).**

Example 6

Convert 0.004 seconds to milliseconds.

SOLUTION

1. "Milli" means 10^{-3}, so $1\ \text{ms} = 0.001\ \text{s}$, i.e. $1\ \text{s} = \boxed{1000}\ \text{ms}$.
2. $0.004 \times \boxed{1000} = 4$.

$0.004\ \text{s} = \boxed{4}\ \text{ms}$

Note: The decimal point is the hinge of the whole system. Multiplying or dividing by ten changes no digit at all, it only slides the point: $34.7 \times 10 = 347$ and $34.7 \div 10 = 3.47$.

decimal decimal point order of magnitude SI prefix

Mental-Math Shortcut

Multiply or divide by powers of ten by sliding the point. $\times 10^n$ moves the decimal point n places right; $\div 10^n$ (i.e. $\times 10^{-n}$) moves it n places left. No long multiplication: $6.02 \times 10^4 = 60{,}200$.

Count digits to judge size. A whole number with d digits is about 10^{d-1}. A seven-digit figure is "millions" (10^6); a ten-digit one is "billions" in scale.

Metric in your head. Neighbouring SI prefixes (kilo, mega, giga; milli, micro, nano) are $1000\times$ apart, so jumping one prefix just moves the point three places.

🧠 The Logic of It

There is nothing mathematically special about ten. A civilisation of eight-fingered beings would prove every theorem we do, simply writing the numbers in base eight. What makes ten *useful to us* is alignment: our counting base equals the base of the metric system and of scientific notation, so unit changes and magnitude estimates collapse into moving a single point. Convenience, not truth, is the whole case for base ten.

⚠ Common Mistakes

- Thinking base ten is "natural" or mathematically necessary. It is a human convention rooted in ten fingers; computers manage without it.
- Miscounting the decimal shift in scientific notation. Moving the point left makes the exponent **positive** for big numbers ($93{,}000{,}000 = 9.3 \times 10^{7}$).
- Mixing up SI prefixes: kilo is 10^{3}, not 10^{2}; milli is 10^{-3}, not 10^{-2} (that is centi).
- Believing a zero is "nothing". A zero is a **place-holder** that keeps the other digits in their correct power-of-ten columns: $407 \neq 47$.
- Reading more digits as automatically "much bigger" without checking the leading digit: $1{,}000{,}000$ and $9{,}000{,}000$ are the same order of magnitude.

Quiz

63.1.Q1. What is the place value of the 5 in $3{,}512$?

A. 5

B. 50

C. 500

D. 5000

63.1.Q2. Write $480{,}000$ in scientific notation.

63.1.Q3. Convert 7 kilometres to metres.

63.1.Q4. Which SI prefix means 10^{-3}?

A. centi

B. milli

C. micro

D. kilo

63.1.Q5. Roughly what order of magnitude is a 9-digit number?

63.1.Q6. Compute 3.6×10^{3}.

63.1.Q7. True or False: humans use base ten because it is mathematically the best base.

A. True

B. False

63.1.Q8. How many minutes are in an hour, and which old base does that come from?

63.1.Q9. Convert 250 milligrams to grams.

🐍 From Concept to Code (Python)

Python writes decimals just as you would, and supports scientific notation with the letter `e` (read "times ten to the"). Underscores may group digits for readability, and format specifiers print numbers in scientific or fixed form.

```
speed = 3.0e8          # 3.0 x 10^8 metres/second
print(speed)
big = 93_000_000       # underscores are ignored, just readable
print(big)
print(f"{big:.2e}")    # scientific notation, 2 decimals
```

```
output
300000000.0
93000000
9.30e+07
```

```
km = 2.5
print(km * 10**3, "metres")  # kilo = 10^3
print(0.004 * 10**3, "ms")   # seconds to milliseconds
```

```
output
2500.0 metres
4.0 ms
```

Try it:

1. Predict, then run: `print(1e3 == 1000)`. Why is the result printed as a float?
2. Print the Sun-to-Earth distance `1.496e11` metres in kilometres by dividing by `1e3`.

Answer Key – Chapter 63 – Base 10: The Decimal System

63.1.Q1. 500 – it sits in the hundreds (10^2) column.

63.1.Q2. 4.8×10^5.

63.1.Q3. 7000 m – kilo is 10^3.

63.1.Q4. milli.

63.1.Q5. About 10^8 (hundreds of millions).

63.1.Q6. 3600 – slide the point three places right.

63.1.Q7. False – it comes from having ten fingers; it is a convention.

63.1.Q8. 60 minutes, from the Babylonian base-sixty system.

63.1.Q9. 0.25 g – milli is 10^{-3}, so divide by 1000.

Chapter 64 · Base 2: Binary & How Computers Count

What binary (base two) is and why every computer runs on it: the on/off world of transistors, how bits group into bytes, how the same 1s and 0s come to mean numbers, letters, colours, sounds and images, and how file formats tell the machine which meaning to use.

The Idea Behind It

Computers count in base two, using only 0 and 1. *but why?*

A computer is built from billions of tiny switches called **transistors**, and a switch has exactly two reliable states: **off** or **on**. Call them 0 and 1 and you have the two digits of base two. There is no third symbol, because there is no "half-on" a circuit can tell apart without error. So the natural base for a machine made of switches is **two**.

$$\text{off} = 0\,, \qquad \text{on} = 1$$

Why is that enough for everything? Because each extra switch **doubles** how many patterns you can make: 1 bit gives 2 states, 2 bits give 4, and n bits give 2^n. That is the same place-value rule as any base, now with $b = 2$:

$$\boxed{n \text{ bits} \;\Rightarrow\; 2^n \text{ distinct patterns}}$$

$\therefore$ from two humble states, 0 and 1, a row of n switches builds 2^n different "words", and those patterns are enough to stand for every number, letter, colour and sound a computer handles.

two states, doubling each step

Everything a computer does, this sentence, your music, a video call, is underneath nothing but **patterns of two symbols**: 0 and 1. That is **binary**, base two. It looks impossibly bare, yet it is the perfect language for a machine, and the rest of this chapter shows how those two symbols come to represent the whole digital world.

BINARY (BASE TWO)

The **binary** system is base 2: just two digits, 0 and 1, each called a **bit** (short for *binary digit*). Reading right to left, the columns are the powers of two: $1, 2, 4, 8, 16, \ldots$

The hardware reason: a world of switches

A modern chip holds **billions of transistors**, each just a microscopic switch that is either **off** or **on** (no current or current; low voltage or high). Two states are easy to build and, crucially, easy to tell apart without error even when signals are noisy. Reliably distinguishing ten voltage levels, to count in base ten, would be far harder and far more fragile. So engineers chose the simplest possible alphabet: **off and on**, written 0 and 1.

$$0 = \text{off (low voltage)}, \qquad 1 = \text{on (high voltage)}$$

Reading binary: place value with powers of two

Binary is read exactly like decimal, but the columns **double** instead of multiplying by ten. Keep the columns that hold a 1 and add them.

$$1011_2 = 1\cdot 8 + 0\cdot 4 + 1\cdot 2 + 1\cdot 1 = 8+2+1 = 11$$

Binary	8	4	2	1	Decimal
0001				1	1
0010			1		2
0101		1		1	5
1010	1		1		10
1111	1	1	1	1	15

From bits to bytes

A single **bit** is the smallest unit of information: one yes/no, one 0 or 1. Bits are bundled for convenience: - A group of 8 bits is a **byte**. (A group of 4, a half-byte, is a *nibble*.) - One byte has $2^8 = \mathbf{256}$ possible patterns, so it can hold a number from 0 to 255. - Bytes group further into kilobytes, megabytes and gigabytes, each step about $1000\times$ larger (precisely $1024 = 2^{10}$ in the binary sense).

> **Note:** Mind the units: a **bit** is one $0/1$; a **byte** is a block of eight bits. Internet speeds are quoted in *bits* per second (Mbps) but file sizes in *bytes* (MB), so an "80 Mbps" line downloads only about 10 MB each second.

Unit	Size	Roughly holds
Bit	1 binary digit	one on/off
Byte	8 bits	one letter; 0 to 255
Kilobyte (KB)	1024 bytes	a short paragraph
Megabyte (MB)	1024 KB	a minute of music
Gigabyte (GB)	1024 MB	a long film

How 1s and 0s become the real world

A pattern of bits means nothing until we agree on an **encoding**: a code book that says which pattern stands for what. Change the code book and the same bits become something else entirely.

Numbers. Whole numbers are stored directly in binary place value, as above. (Cleverer schemes called *two's complement* and *floating point* extend this to negative and decimal numbers, but the core is still powers of two.)

Letters and text. A code book maps each character to a number, which is then stored as bits. The classic one is **ASCII**: for example A is 65, B is 66, and a space is 32. So "Hi" is two bytes, 72 and 105.

$$\texttt{A} = 65 = 0100\,0001_2, \qquad \texttt{H} = 72, \quad \texttt{i} = 105$$

Modern text uses **Unicode** (via UTF-8), which extends the same idea to every alphabet and to emoji.

Colours. A screen pixel mixes red, green and blue light. Give each channel one byte (0 to 255) and you can describe about 16.7 million colours (256^3). Programmers usually write those three bytes as a six-digit **hex** code: pure red is `#FF0000` (red 255, green 0, blue 0), and white is `#FFFFFF`.

Sound. A microphone measures air pressure thousands of times a second, and each measurement is saved as a number (a *sample*). Played back fast enough (say $44{,}100$ samples a second), those numbers recreate the wave your ear hears.

Images. A photo is a grid of pixels, each pixel three colour bytes. A 1000×1000 image therefore needs about $1000 \cdot 1000 \cdot 3 = 3$ million bytes (3 MB) before compression.

File formats: the same bits, different meanings

Here is the deep point. A drive only ever stores bytes; it has no idea whether a given run of bytes is a poem, a photo or a song. The **file format** is the agreed code book, and the **extension** (`.docx`, `.jpg`, `.mp3`, `.mp4`) plus a hidden header (a "magic number" at the start of the file) tell the computer *which* code book to use: - `.docx` is a zipped bundle of XML describing text, styles and layout. - `.jpg` (or `.jpeg`) stores pixels with clever compression that drops detail the eye barely notices. - `.mp3` stores sound samples compressed by discarding sounds you cannot easily hear. - `.mp4` is a *container* holding compressed video alongside its audio.

This is why **renaming** a file does not convert it: calling `song.mp3` by the name `song.jpg` rearranges not one byte, so an image viewer opens it, follows the wrong code book, and sees only garbage. Real conversion has to *re-encode* the bytes under the new format's rules.

> **Tip:** The first few bytes of a file are usually a "magic number" identifying the format whatever the name says: every PNG begins with the same 8 signature bytes, every PDF with the characters `%PDF`.

Worked examples

Example 1

Convert 1101_2 to decimal.

SOLUTION

1. Columns are $8\ \ 4\ \ 2\ \ 1$ and the digits are $1, 1, 0, 1$.
2. Add the columns holding a 1: $8 + 4 + 0 + 1$.

$1101_2 = 13$

Example 2

Convert 25_{10} to binary.

SOLUTION

1. Find the powers of two that fit: 16 fits ($25 - 16 = 9$), then 8 ($9 - 8 = 1$), then 1.
2. Put a 1 in the 16, 8 and 1 columns and 0 elsewhere: $16 + 8 + 0 + 0 + 1$.

$25_{10} = 11001_2$ **(check:** $16 + 8 + 1 = 25$**)**

Example 3

How many different values can one byte (8 bits) hold?

SOLUTION

1. Each bit doubles the count, so 8 bits give 2^8.
2. $2^8 = 256$ patterns, numbered 0 through 255.

256 **values (0 to 255)**

Example 4

The letter A is ASCII 65. Write it as a byte in binary, then decode 01000010_2.

SOLUTION

1. $65 = 64 + 1 = 1000001_2$, padded to a byte as 01000001.
2. $01000010_2 = 64 + 2 = 66$, one past 65, so it is the letter B.

A $= 01000001_2$**;** $01000010_2 = 66$ **= B**

Example 5

A colour is written `#00FF00`. What colour is it?

SOLUTION

1. Split into three bytes: red $= 00$, green $=$ FF, blue $= 00$.
2. $\mathrm{FF}_{16} = 255$ (full) while the others are 0 (none): all green light.

Pure green**.**

Example 6

About how many bytes does a 200×300-pixel photo need uncompressed, at 3 bytes per pixel?

SOLUTION

1. Count the pixels: $200 \times 300 = 60{,}000$.
2. Three colour bytes each: $60{,}000 \times 3 = 180{,}000$.

$180{,}000$ **bytes, about** 176 **KB.**

Example 7

Your connection is 80 Mbps (megabits per second). Roughly how many megabytes does it download per second?

SOLUTION

1. Speed is in **bits** but files are in **bytes**, and 1 byte $= 8$ bits.
2. Divide by 8: $80 \div 8 = 10$.

About 10 MB per second.

Example 8

Add the binary numbers $101_2 + 11_2$.

SOLUTION

1. Line them up and add column by column, carrying when a column reaches 2 (which is 10_2).
2. $1 + 1 = 10$ (write 0, carry 1); then $0 + 1 + 1 = 10$ (write 0, carry 1); then $1 + 1 = 10$.
3. Read the result 1000_2, and check in decimal: $5 + 3 = 8$.

$101_2 + 11_2 = 1000_2$ **(which is 8)**

Note: Inside the machine there are no numbers, letters or colours, only switches that are off or on. All "meaning" lives in the encoding and file format we agree to lay over the bits.

binary bit byte transistor ASCII encoding file format

⚡ **Mental-Math Shortcut**

Memorise the powers of two: $1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024$. Reading binary then becomes instant, just add the column values under each 1.

Doubling and halving. Multiplying a binary number by 2 shifts every bit one place left and adds a 0 on the end (just as $\times 10$ adds a zero in decimal). Halving shifts right.

Byte ranges by heart. 1 byte covers 0 to 255; 2 bytes cover 0 to $65{,}535$; each added byte multiplies the range by 256. And remember to divide a "bits per second" speed by 8 to get bytes.

🧠 **The Logic of It**

Binary and the logic of Chapter 17 onward are the *same subject* in different clothes. A bit is a truth value: 1 is **true**, 0 is **false**. The transistors are wired into **logic gates** (AND, OR, NOT) that compute exactly the truth tables you already drew, and stacks of those gates do arithmetic, comparisons and decisions. A computer is, quite literally, Boolean logic made of electricity: the on/off switch is the physical form of true/false.

⚠ Common Mistakes

- Confusing a **bit** with a **byte**. A bit is one $0/1$; a byte is eight of them, so "8 megabits" is only 1 megabyte.
- Forgetting binary columns are powers of **two**, not ten: 10_2 is 2, not ten, and 100_2 is 4.
- Assuming the file extension changes the contents. Renaming `.mp3` to `.jpg` rewrites no bytes; only re-encoding actually converts a file.
- Thinking bits "are" letters or colours on their own. The same byte 65 is the number 65, the letter `A`, or a shade of grey, depending on the encoding in force.
- Mixing 1000 and 1024. In the binary sense a kilobyte is $2^{10} = 1024$ bytes, not exactly 1000.

Quiz

64.1.Q1. What is 110_2 in decimal?

A. 6

B. 3

C. 110

D. 4

64.1.Q2. How many bits are in a byte?

A. 4

B. 8

C. 16

D. 256

64.1.Q3. Convert 19_{10} to binary.

64.1.Q4. How many distinct values can 2 bytes (16 bits) represent?

64.1.Q5. In ASCII A is 65, so what number is C?

A. 66

B. 67

C. 99

D. 3

64.1.Q6. The hex colour `#0000FF` is which colour?

A. red

B. green

C. blue

D. white

64.1.Q7. True or False: renaming `photo.png` to `photo.mp3` turns it into playable music.

A. True

B. False

64.1.Q8. Why do computers use base two rather than base ten?

64.1.Q9. Add $11_2 + 1_2$ in binary.

64.1.Q10. A file begins with the bytes `%PDF`. What does that tell the computer?

🐍 From Concept to Code (Python)

Python lets you see the bits directly. `bin(n)` shows a number in binary, `ord()` and `chr()` cross the letter/number bridge (the ASCII/Unicode code), and the bitwise operators `& | ^ << >>` work on the underlying bits, the very operations the hardware performs.

```
print(bin(13))          # number to binary
print(int("11001", 2))  # binary text to number
print(ord("A"))         # letter to its code
print(chr(66))          # code back to a letter
```

```
output
0b1101
25
65
B
```

```
print(0xFF)             # a hex byte
print(5 << 1)           # shift left = x2  (101 -> 1010)
print(0b1100 & 0b1010)  # bitwise AND
r, g, b = 255, 0, 0
print(f"#{r:02X}{g:02X}{b:02X}")  # build a hex colour
```

```
output
255
10
8
#FF0000
```

Try it:

1. Predict, then run: `print(bin(255))`. How many 1s do you expect, and why?
2. Loop over `"Hi"` printing `ord(c)` for each character to see the bytes that spell it.
3. Encode a string to raw bytes with `"Hi".encode("utf-8")` and look at the result.

Answer Key – Chapter 64 – Base 2: Binary & How Computers Count

64.1.Q1. 6 – columns $4 + 2 + 0$.

64.1.Q2. 8.

64.1.Q3. 10011_2 – it is $16 + 2 + 1$.

64.1.Q4. $2^{16} = 65{,}536$ (numbers 0 to $65{,}535$).

64.1.Q5. 67 – the letters are consecutive: A, B, C are $65, 66, 67$.

64.1.Q6. blue – only the blue byte is full ($\mathrm{FF} = 255$).

64.1.Q7. False – the bytes are unchanged; the wrong code book is simply applied.

64.1.Q8. Their switches (transistors) have two reliable states, off and on, which map directly onto 0 and 1.

64.1.Q9. 100_2 – $3 + 1 = 4$; the carries ripple left.

64.1.Q10. It is a magic number identifying the file as a PDF, so the PDF code book is used whatever the name says.



PART XII

Python: From Concept to Code

You already understand the ideas – now meet the language. Every concept from this book, gathered into a hands-on path you can run on a real computer.

Chapter 65 – Getting Started: the REPL & print() · Chapter 66 – Variables & Types · Chapter 67 – Operators & Precedence · Chapter 68 – Booleans, Comparisons & Conditionals · Chapter 69 – Loops · Chapter 70 – Functions · Chapter 71 – Collections: Lists, Sets & Dictionaries · Chapter 72 – Putting It Together: Small Programs · Chapter 73 – Where to Go Next

Chapter 65 · Getting Started: the REPL & print()

> *How to run Python, type expressions into the interactive prompt, print results with print(), leave notes with comments, and watch Python obey the very same order of operations you learned in Chapter 1.*

The Idea Behind It

Typing code into the prompt is evaluating an expression, exactly like simplifying a sum on paper.

but why?

When you simplify $2 + 3 \times 4$ by hand you do not change the symbols, you *reduce* them: you turn a written expression into the single **value** it stands for. The REPL does the identical thing, in three beats: it **reads** what you type, **evaluates** it to one value, and **prints** that value back. Input becomes output by collapsing the expression, just as on paper.

$$\underbrace{2 + 3 \times 4}_{\text{expression}} \longrightarrow 2 + 12 \longrightarrow 14$$

So the prompt is not a new kind of arithmetic, it is the *same* arithmetic with a faithful machine doing the reducing. Type the expression, and the value it denotes comes straight back:

```
>>> 2 + 3 * 4
14
```

$$\therefore \text{ code at the prompt } = \text{ an expression evaluated to its value}$$

That single loop, read then evaluate then print, is the whole engine under every program in this book.

you already know how to simplify, Python just does it faster

You have spent this whole book building the *ideas*: numbers, order of operations, logic, algebra. Here is the wonderful part – Python already understands almost all of it. In this final part we meet the language itself, one small step at a time, and you will recognise nearly everything. If you can do the arithmetic, you can already read the code.

The REPL: a calculator that talks back

The fastest way to meet Python is the **REPL** – short for *Read–Eval–Print Loop*. It **reads** what you type, **evaluates** it, **prints** the answer, and **loops** back for more. You start it by running the command `python` (or `python3`) in a terminal; the prompt >>> means Python is waiting for you.

Type an expression, press Enter, and the answer appears immediately – no "print" needed at the prompt, because the REPL shows the value of whatever you type:

```
>>> 2 + 3
5
>>> 7 * 8
56
>>> 100 - 1
99
```

That is it – Python is a calculator that talks back. Everything else in this book is built on top of this simple loop.

print(): asking Python to show something

Inside a saved program (a file ending in `.py`), typing `2 + 3` is **not** enough – the program computes the answer but keeps it to itself. To make a result appear on screen you use the **`print()`** function: you put what you want shown inside the parentheses.

```
print(2 + 3)
print(7 * 8)
print("Hello, world!")
```

```
output
5
56
Hello, world!
```

Two things to notice. First, `print` is followed by **parentheses** holding its input – just like $f(x)$ in Chapter 29 held *its* input. Second, the words `"Hello, world!"` sit inside **quotation marks**: that makes them a piece of **text** (a *string*) to be shown exactly, rather than something to calculate. We will meet strings properly in the next chapter.

You can print several things at once by separating them with commas; Python puts a single space between them:

```
print("The answer is", 6 * 7)
```

```
output
The answer is 42
```

Comments: notes Python ignores

A **comment** starts with a #. Everything after the # on that line is ignored by Python – it is a note for *humans* reading the code. Good comments explain *why* you did something, not the obvious *what*.

```
# This line is just a note and does nothing.
print(10 / 2)   # you can also comment at the end of a line
# print(999)    <- a "commented-out" line never runs
```

```
output
```

```
5.0
```

Note: Notice the answer came out as <code>5.0</code>, not <code>5</code> – in Python the division sign <code>/</code> always produces a decimal. That tiny detail is exactly the kind of thing we will explore in Chapter 32.

Order of operations – already built in

Remember the very first chapter, BODMAS / PEMDAS? Python obeys the **identical** order of operations, calling it *operator precedence*. Powers (written ∗∗) come first, then ∗ and /, then + and −, left to right – and parentheses override everything, exactly as on paper.

```
print(3 + 4 * 2)        # multiply first, then add
print((3 + 4) * 2)      # parentheses force the addition first
print(5 + 2 ** 3 * 4)   # power, then multiply, then add
```

```
output
11
14
37
```

Trace that last line the way you would by hand: $2^3 = 8$, then $8 \times 4 = 32$, then $5 + 32 = 37$. Python reached the same answer by the same rules. You are not learning new arithmetic – you are learning to *type* the arithmetic you already know.

Because each `print()` ends with a fresh line, several of them stack up a tidy result on screen. Here Python works out three expressions from Chapter 1 and labels each answer with a comma:

```
print("12 plus 8 times 2 =", 12 + 8 * 2)
print("(12 plus 8) times 2 =", (12 + 8) * 2)
print("half of 50 =", 50 / 2)
```

```
output
12 plus 8 times 2 = 28
(12 plus 8) times 2 = 40
half of 50 = 25.0
```

Notice the brackets flipped the answer from 28 to 40, exactly as they would on paper, and the last line came out as `25.0` because / always gives a decimal. The REPL shows the same thing without any `print()` – type a bare expression and its value appears, so you can check a sum from the book in seconds:

```
>>> 9 + 6 / 3
11.0
>>> (9 + 6) / 3
5.0
>>> 2 ** 4
16
```

Tip: Keep a REPL window open while you read these chapters. Type every example yourself, then change a number and predict the new answer before pressing Enter. Nothing builds fluency faster than that little guessing game.

Errors are normal

When you mistype, Python prints an **error message** instead of an answer. This is not a scolding – it is a helpful note pointing at the problem. Forgetting a closing quote or parenthesis is the most common slip when you start:

```
print("hello)
```

```
output
  File "<stdin>", line 1
    print("hello)
          ^
SyntaxError: unterminated string literal
```

Read the message bottom-up: the **last** line names the kind of problem (`SyntaxError`), and the little `^` points near where Python got confused. Fix the missing quote and it runs. Every programmer reads errors like this all day long – it is a skill, not a failure.

Quiz

65.1.Q1. What does **REPL** stand for?

A. Read–Eval–Print Loop

B. Run–Edit–Print Line

C. Read–Enter–Process Log

D. Repeat–Execute–Print–Leave

65.1.Q2. Predict the output: `print(6 * 7)`

65.1.Q3. Predict the output: `print(3 + 4 * 2)`

A. `14`

B. `11`

C. `10`

D. `24`

65.1.Q4. Predict the output: `print((3 + 4) * 2)`

65.1.Q5. What appears when Python runs this line? `print(10 / 2)`

A. `5`

B. `5.0`

C. `10/2`

D. an error

65.1.Q6. What does a line beginning with # do?

A. Prints a number sign

B. Nothing – it is a comment Python ignores

C. Causes an error

D. Repeats the previous line

65.1.Q7. Predict the output: `print("The answer is", 5 + 5)`

65.1.Q8. True or False: at the >>> REPL prompt, typing `2 + 2` and pressing Enter shows `4` without needing `print()`.

A. True

B. False

Answer Key – Chapter 65 – Getting Started: the REPL & print()

65.1.Q1. Read–Eval–Print Loop – it reads input, evaluates it, prints the result, and loops.

65.1.Q2. 42 – Python evaluates `6 * 7` and prints the result.

65.1.Q3. 11 – multiplication before addition (4∗2=8, then 3+8), exactly like Chapter 1.

65.1.Q4. 14 – parentheses force the addition first (7), then 7∗2.

65.1.Q5. `5.0` – the `/` operator always gives a decimal (float) result.

65.1.Q6. Nothing – everything after # is a comment for humans and is ignored by Python.

65.1.Q7. `The answer is 10` – commas print several items separated by a single space.

65.1.Q8. True – the REPL automatically prints the value of whatever you type.

Chapter 66 · Variables & Types

How to store values in named variables, the four everyday types Python uses (int, float, str, bool), how to check a value's type, and the simple rules for naming things well.

The Idea Behind It

`x = x + 1` is nonsense as an equation but perfect as an instruction. *but why?*

Read $=$ the way algebra trained you, *"the two sides are the same number"*, and $x = x + 1$ says x equals x plus one, which is impossible: subtract x from both sides and you get the falsehood $0 = 1$. Yet the line runs happily in Python. The resolution is that Python's $=$ is **not** the equals of algebra at all.

$$\underbrace{x = x + 1}_{\text{as an equation}} \;\Rightarrow\; 0 = 1 \qquad (\text{false})$$

Python's $=$ means *store*: work out the right-hand side using the value x **has now**, then put that result back in the box labelled x. It is a step in time, not a timeless claim. With x starting at 5:

$$x = x + 1 \;\longrightarrow\; 5 + 1 \;\longrightarrow\; x \text{ now holds } 6$$

```
x = 5
x = x + 1
print(x)
```

```
output
6
```

$$\therefore\ (=) \text{ means } \boxed{\text{store}}\text{, not } \textit{is equal to}$$

A variable is a labelled box, and assignment refills it.

to test equality Python uses a different symbol, ==, which you meet in Chapter 33

In algebra you wrote $x = 4$ and then used x wherever you meant that number. Python does exactly the same thing, and calls x a **variable** – a name that holds a value so you can use it again later. This one idea, *give it a name, reuse the name*, is the backbone of every program ever written.

Assignment: giving a value a name

You create a variable with the **assignment** operator `=`. Read it not as "equals" but as *"gets"* or *"is set to"*: the name on the **left** is set to the value on the **right**.

```
age = 20
price = 4.50
name = "Ada"
print(age)
print(price)
print(name)
```

```
output
20
4.5
Ada
```

Once a name holds a value, you can compute with it just like a number in algebra. Change the value in one place and every later use follows along:

```
width = 5
height = 3
area = width * height
print(area)
```

```
output
15
```

Naming the parts of a formula makes it read like the maths it came from. Here we store a Celsius temperature, then apply the conversion rule $F = \frac{9}{5}C + 32$ to get Fahrenheit:

```
celsius = 20
fahrenheit = 9 / 5 * celsius + 32
print(fahrenheit)
```

```
output
68.0
```

Note: In maths, $=$ states a fact that is *already* true. In Python, <code>=</code> is an *instruction*: "make the left-hand name refer to the right-hand value." That is why <code>area = width * height</code> works – Python first works out the right side, then pins the answer to the name <code>area</code>.

Reassignment: names can change

A variable can be given a **new** value at any time; the old one is simply forgotten. The right-hand side is always worked out *first*, using the current values, so a line like `score = score + 10` makes perfect sense even though it would be nonsense in algebra:

```
score = 0
score = score + 10     # take the current score (0), add 10, store it back
score = score + 5
print(score)
```

```
output
15
```

Tip: Because adding to a variable is so common, Python lets you shorten <code>score = score + 10</code> to <code>score += 10</code>. The shortcuts <code>-=</code>, <code>*=</code> and <code>/=</code> work the same way.

The four everyday types

Every value in Python has a **type** – what *kind* of thing it is. Four types cover almost everything you will do at first:

- **`int`** – a whole number (an *integer*), like the $\mathbb{Z}$ of Chapter 6: `20`, `0`, `-7`.
- **`float`** – a number with a decimal point (a *floating-point* number): `4.5`, `3.0`, `-0.25`.
- **`str`** – a *string* of text, always written inside quotes: `"Ada"`, `"hello"`, `"42"`.
- **`bool`** – a *Boolean* truth value from Part V: `True` or `False` (note the capital letters).

You can ask Python what type any value has with the built-in **`type()`**:

```
print(type(20))
print(type(4.5))
print(type("Ada"))
print(type(True))
```

```
output
<class 'int'>
<class 'float'>
<class 'str'>
<class 'bool'>
```

The type matters because it decides what an operation *means*. With numbers, `+` adds; with strings, `+` joins text end to end (this is called *concatenation*):

```
print(3 + 4)
print("3" + "4")
print("Ada" + " Lovelace")
```

```
output
```

```
7
34
Ada Lovelace
```

Notice `"3" + "4"` gave `34`, not `7` – those are *strings of text*, not numbers, so Python glued them together. Keeping numbers and text straight is one of the first habits a programmer builds.

Turning one type into another

You can **convert** between types with `int()`, `float()` and `str()`. This matters most when reading input, which always arrives as text:

```
x = "12"          # text, not a number
n = int(x)        # convert the text to an integer
print(n + 8)
print(str(2025) + " edition")
```

```
output
20
2025 edition
```

Converting also lets you average a few values once they are real numbers. Suppose two test scores arrive as the text `"80"` and `"90"`; turn each into an `int`, then take their mean – add them up and divide by how many there are:

```
a = int("80")
b = int("90")
mean = (a + b) / 2
print("Average:", mean)
```

```
output
Average: 85.0
```

Naming things well

A variable name may use letters, digits and underscores, but must **not start with a digit** and must not contain spaces. Names are *case-sensitive*: `age` and `Age` are different. The golden rule: choose names that say what the value *means*.

- Good: `total_price`, `student_count`, `is_prime`
- Avoid: `x2`, `tp`, `thing` (clear only to you, today)

```
days = 7
hours_per_day = 24
total_hours = days * hours_per_day
print(total_hours)
```

```
output
168
```

Code is read far more often than it is written – usually by *you*, weeks later. A good name is a gift to your future self.

Quiz

66.1.Q1. After `x = 5` then `x = x + 3`, what does `print(x)` show?

A. `5`

B. `8`

C. `x + 3`

D. an error

66.1.Q2. What is the type of the value `4.5`?

A. `int`

B. `float`

C. `str`

D. `bool`

66.1.Q3. Predict the output: `print("2" + "3")`

A. `5`

B. `23`

C. `"23"`

D. an error

66.1.Q4. Predict the output: `print(type(True))`

66.1.Q5. Which is a valid variable name: `2cats`, `total_cats`, `total cats`, or `total-cats`?

66.1.Q6. Predict the output: `n = int("15"); print(n + 5)`

66.1.Q7. True or False: `age` and `Age` refer to the same variable.

A. True

B. False

66.1.Q8. Rewrite `count = count + 1` using a shortcut operator.

Answer Key – Chapter 66 – Variables & Types

66.1.Q1. `8` – the right side `5 + 3` is worked out first, then stored back into `x`.

66.1.Q2. `float` – it has a decimal point.

66.1.Q3. `23` – these are strings, so `+` joins the text (concatenation), it does not add.

66.1.Q4. `<class 'bool'>` – `True` is a Boolean value.

66.1.Q5. `total_cats` – names cannot start with a digit, contain a space, or use a hyphen.

66.1.Q6. `20` – `int("15")` converts the text to the number 15, then `15 + 5`.

66.1.Q7. False – Python names are case-sensitive, so they are two different variables.

66.1.Q8. `count += 1` – the `+=` operator adds to a variable in place.

Chapter 67 · Operators & Precedence

The full set of Python arithmetic operators – including the two kinds of division and the power operator – how precedence mirrors Chapter 1, and how int and float values mix.

The Idea Behind It

Floor division `//` and remainder `%` are two halves of one fact: $a = (a//b) \cdot b + (a\%b)$. *but why?*

When you divide a by b the old way you get a *quotient* and a *remainder*. Chapter 5 fixed exactly what those mean: the quotient q counts how many whole bs fit inside a, and the remainder r is the little bit left over, always smaller than b. That is the **division identity**, the law every divide obeys:

$$\boxed{a = q\,b + r, \qquad 0 \le r < b}$$

Python simply gives each piece its own operator: `a // b` returns the quotient q (floor division throws the leftover away), and `a % b` returns the remainder r. They are not two unrelated tricks, they are the two terms of one equation. Take $a = 17$, $b = 5$: how many 5s fit in 17? Three, with 2 to spare.

$$17 = \boxed{3} \cdot 5 + \boxed{2}, \qquad 17//5 = \boxed{3}, \quad 17\,\%\,5 = \boxed{2}$$

Substitute those back into the identity and rebuild a from its parts:

$$(17//5) \cdot 5 + (17\,\%\,5) = \boxed{3} \cdot 5 + \boxed{2} = 15 + 2 = \boxed{17}$$

```
print(17 // 5, 17 % 5)
print((17 // 5) * 5 + (17 % 5) == 17)
```

```
output
3 2
True
```

$$\therefore\ a = (a//b) \cdot b + (a\,\%\,b) = \boxed{a}, \text{ always}$$

That is why `%` is the divisibility test: a remainder of 0 means b goes in exactly, with nothing left over.

quotient and remainder are one fact wearing two hats

You already met `+`, `−`, `*` and `/` at the prompt. Python has a few more arithmetic **operators**, and together they let you compute anything from this book directly in code. Best of all, they follow the **exact order of operations** from Chapter 1 – Python just calls it *operator precedence*.

The arithmetic operators

Here is the complete everyday set:

- `+` add, `−` subtract, `*` multiply – exactly as you expect.
- `/` **true division** – always gives a `float` (a decimal), even when it divides evenly.
- `//` **floor division** – divides and throws away the remainder, giving the whole-number quotient.
- `%` **modulo** – gives the *remainder* after division.

- `**` **power** – raises to an exponent, the `a^{n}` of Chapter 10.

```
print(7 + 2)
print(7 - 2)
print(7 * 2)
print(7 / 2)
print(7 // 2)
print(7 % 2)
print(7 ** 2)
```

```
output
9
5
14
3.5
3
1
49
```

Look closely at the division trio. From Chapter 5 you know $7 \div 2 = 3$ remainder 1. Python shows all three views of that single fact: `7 / 2` is `3.5` (the exact decimal), `7 // 2` is `3` (the quotient), and `7 % 2` is `1` (the remainder). The quotient and remainder together are what long division gives you.

Note: <code>//</code> and <code>%</code> are not just curiosities. <code>n % 2 == 0</code> tests whether <code>n</code> is even (remainder 0) – the divisibility idea from Chapter 5. We will use <code>%</code> constantly when we hunt for factors and primes in Chapter 34.

Floor division and modulo in action

Suppose you have 30 eggs and pack them in boxes of 12. How many full boxes, and how many eggs left over?

```
eggs = 30
boxes = eggs // 12     # how many full boxes
left = eggs % 12       # how many eggs left over
print(boxes, "full boxes,", left, "left over")
```

```
output
2 full boxes, 6 left over
```

Python even has a built-in `divmod()` that hands you both at once, as a pair:

```
print(divmod(30, 12))
```

```
output
(2, 6)
```

Modulo is also how clocks "wrap around". On a 12-hour clock, 5 hours after 9 o'clock is not 14 but 2 – because $14 \bmod 12 = 2$. The % operator does that wrap for you:

```
start = 9
later = (start + 5) % 12
print(later, "o'clock")
```

```
output
2 o'clock
```

The power operator

The ** operator is repeated multiplication – the exponents of Chapter 10. It handles square roots too, since a square root is the power $\frac{1}{2}$:

```
print(2 ** 10)        # 2 to the 10th
print(5 ** 0)         # anything to the 0 is 1
print(2 ** -1)        # negative exponent -> a fraction
print(9 ** 0.5)       # the square root of 9
```

```
output
1024
1
0.5
3.0
```

Squaring is just the power 2, so the area of a square is one short expression. Here we find the area of a square with side 7, then add Pythagoras from Chapter 46: the hypotenuse of a 3–4 right triangle is $\sqrt{3^2 + 4^2}$:

```
side = 7
print("area:", side ** 2)
print("hypotenuse:", (3 ** 2 + 4 ** 2) ** 0.5)
```

```
output
area: 49
hypotenuse: 5.0
```

Precedence: Chapter 1, written in code

When several operators meet, Python ranks them in this order (highest first), then works **left to right** among equals – identical to BODMAS / PEMDAS:

( ) ⇒ ** ⇒ * / // % ⇒ + -

```
print(2 + 3 * 4)        # times before plus
print((2 + 3) * 4)      # parentheses first
print(2 ** 3 ** 2)      # ** groups right-to-left: 2 ** (3 ** 2) = 2 ** 9
print(20 - 6 / 2)       # divide first, then subtract
```

```
output
14
20
512
17.0
```

When in doubt, add parentheses. They cost nothing and make your intention unmistakable to the next reader – even when Python would have got the order right anyway.

Mixing ints and floats

If an expression mixes a whole number and a decimal, Python quietly promotes the result to a `float`, so no precision is lost:

```
print(4 + 1.0)      # int + float -> float
print(10 / 5)       # / always gives a float
print(10 // 3)      # // of two ints stays an int
print(3 * 2.5)
```

```
output
5.0
2.0
3
7.5
```

Tip: If you ever need a whole number back from a float, wrap it in <code>int(...)</code> – but remember it simply chops off the decimal part: <code>int(7.9)</code> is <code>7</code>, not <code>8</code>. To round properly, use <code>round(7.9)</code>.

Quiz

67.1.Q1. Predict the output: `print(17 // 5)`

A. `3.4`

B. `3`

C. `4`

D. `2`

67.1.Q2. Predict the output: `print(17 % 5)`

67.1.Q3. Predict the output: `print(2 ** 5)`

A. `10`

B. `25`

C. `32`

D. `7`

67.1.Q4. Predict the output: `print(10 / 2)`

A. `5`

B. `5.0`

C. `2`

D. an error

67.1.Q5. Predict the output: `print(2 + 3 * 4)`

67.1.Q6. Which test checks whether `n` is even?

A. `n / 2 == 0`

B. `n % 2 == 0`

C. `n // 2 == 0`

D. `n ** 2 == 0`

67.1.Q7. Predict the output: `print(divmod(23, 4))`

67.1.Q8. True or False: `int(7.9)` gives `8`.

A. True

B. False

Answer Key – Chapter 67 – Operators & Precedence

67.1.Q1. 3 – floor division keeps only the whole-number quotient.

67.1.Q2. 2 – modulo gives the remainder of $17 \div 5$.

67.1.Q3. 32 – `**` is the power operator: $2^5 = 32$.

67.1.Q4. `5.0` – true division `/` always produces a float.

67.1.Q5. 14 – multiplication before addition, exactly as in Chapter 1.

67.1.Q6. `n % 2 == 0` – an even number leaves remainder 0 when divided by 2.

67.1.Q7. `(5, 3)` – 23 = 4 × 5 + 3, so the quotient is 5 and the remainder is 3.

67.1.Q8. False – `int()` chops off the decimal, giving 7. Use `round(7.9)` for 8.

Chapter 68 · Booleans, Comparisons & Conditionals

How to ask Python true/false questions with comparisons, combine them with and/or/not, and make your program choose a path with if/elif/else.

The Idea Behind It

Why `False and X` can skip X entirely (short-circuiting) and still be correct. *but why?*

Python's `and`, `or` and `not` are not new ideas, they are the connectives $\wedge$, $\vee$, $\neg$ from Part V, defined by the very same truth tables. So the truth table *is* the law. Look at the row of $\wedge$ (`and`) where the left side is already **False**:

P	Q	P and Q
F	T	F
F	F	F

Both rows agree: when P is **False**, $P \wedge Q$ is **False** *no matter what Q is*. The value of Q never changes the answer, so the answer is settled the instant the left side is false.

$$\text{False} \wedge Q = \text{False} \quad \text{for every } Q$$

That licences a shortcut: Python reads left to right, and the moment `and` sees a false left side it returns `False` **without even evaluating** the right side. (Symmetrically, `True or X` returns `True` unread.) The skipped work cannot have mattered, because the truth table already decided.

```
def boom():
    print("ran!")
    return True

print(False and boom())
```

```
output
False
```

$$\therefore \text{ short-circuit is valid: the law fixes the result before } Q \text{ is read}$$

Notice `boom()` never printed, proof that Python skipped it. Code booleans are logical connectives, so the truth table you trust is the rule Python runs on. *logic on paper, logic in the machine, same rules*

Back in Part V you learned about **statements** that are either true or false, and the connectives **AND**, **OR** and **NOT** that join them. Python has all of this built in, and it uses the very same logic. The only difference is that here you can *run* it and watch the answers appear.

A value that is either true or false is called a **boolean**. In Python the two boolean values are written `True` and `False` – capital first letter, no quotes. They are real values, just like numbers, so you can store them in variables and print them.

Asking a question: comparison operators

A *comparison* compares two values and hands you back a boolean. These are the same symbols you met on the number line and in algebra:

```
print(3 < 5)       # is 3 less than 5?
print(10 == 10)    # equal?  (two equals signs!)
print(7 != 4)      # not equal?
print(2 >= 9)      # greater than or equal?
```

```
output
True
True
True
False
```

Note: Use a single $=$ to *assign* a value to a variable, and a double $==$ to *test* whether two things are equal. Mixing them up is the most common beginner slip of all. Think: one equals "make it so", two equals "is it so?"

Here are all six comparisons side by side with the math you already know:

Python	Math	Reads as
==	$=$	equal to
!=	$\neq$	not equal to
<	$<$	less than
>	$>$	greater than
<=	$\leq$	less than or equal to
>=	$\geq$	greater than or equal to

Combining questions: and, or, not

To join comparisons, Python uses the plain English words **and**, **or** and **not** – these are exactly the connectives $\wedge$, $\vee$ and $\neg$ from your logic chapters.

```
age = 20
print(age >= 18 and age < 65)   # both must be true
print(age < 13 or age > 65)     # either one true is enough
print(not age == 20)            # flips True to False
```

```
output
True
False
False
```

Just like a truth table: `and` is only `True` when **both** sides are true; `or` is `True` when **at least one** side is true; `not` flips a boolean to its opposite.

You can make Python *print* a truth table to see the rule for yourself. This walks through every combination of two booleans and shows what `and` gives – exactly the four rows you drew in Chapter 19:

```
for p in (True, False):
    for q in (True, False):
        print(p, q, "->", p and q)
```

```
output
True True -> True
True False -> False
False True -> False
False False -> False
```

Truthiness: what counts as True?

Python lets you test almost any value as if it were a boolean. The rule is simple. These are treated as **False**: the number `0`, an empty string `""`, and an empty list `[]`. Almost everything else – any non-zero number, any text with characters in it – is treated as **True**.

```
print(bool(0))      # zero is falsey
print(bool(42))     # any other number is truthy
print(bool(""))     # empty text is falsey
print(bool("hi"))   # non-empty text is truthy
```

```
output
False
True
False
True
```

Making a decision: if / elif / else

Now the payoff. An **if statement** runs a block of code *only when* a condition is true. This is the implication "if p then q" from Chapter 20, turned into action. Notice two pieces of Python grammar: the line ends with a **colon `:`**, and the code that belongs to the `if` is **indented** (four spaces) underneath it.

```
temp = 30
if temp > 25:
    print("It is warm today")
```

```
output
It is warm today
```

Add **else** to say "otherwise, do this instead", and **elif** (short for "else if") to test more conditions in turn. Python checks them top to bottom and runs the **first** one that is true, then skips the rest.

```
score = 72
if score >= 80:
    print("A grade")
elif score >= 60:
    print("B grade")
else:
    print("Keep practising")
```

```
output
B grade
```

The score was 72. Python checked `score >= 80` (false), then `score >= 60` (true) and printed that branch, then stopped – it never reached the `else`.

Putting it together

Conditions can be as rich as you like. Here we combine a comparison with `and` to check whether a year number sits inside a range:

```
year = 2025
if year >= 2000 and year <= 2099:
    print("21st century")
else:
    print("Some other century")
```

```
output
21st century
```

A condition can use the `%` operator from Chapter 32. Recall that `n % 2 == 0` is true exactly when n is even (the divisibility test of Chapter 5). That single comparison lets an `if` sort a number into even or odd:

```
n = 7
if n % 2 == 0:
    print(n, "is even")
else:
    print(n, "is odd")
```

```
output
7 is odd
```

Note: Indentation is not just for looks in Python – it is how the language knows which lines belong to the `if`. Every line in a block must be indented the same amount. A misaligned line will raise an `IndentationError`.

Tip: You can chain comparisons the way you would in maths. Instead of `year >= 2000 and year <= 2099`, Python lets you write `2000 <= year <= 2099`, which reads just like $2000 \leq \text{year} \leq 2099$.

Quiz

68.1.Q1. What does `print(5 == 5)` display?

A. `5`

B. `True`

C. `False`

D. `==`

68.1.Q2. Predict the output: `print(3 > 1 and 4 < 2)`.

68.1.Q3. Predict the output: `print(not (10 != 10))`.

68.1.Q4. Which operator *assigns* a value, and which *tests* for equality?

68.1.Q5. True or False: the value `0` is treated as `True` in a condition.

A. True

B. False

68.1.Q6. Given `n = 50`, predict what this prints: `if n > 100: print("big") elif n > 10: print("medium") else: print("small")`

68.1.Q7. What punctuation must end the line `if temp > 25` for the code to be valid?

A. A semicolon `;`

B. A colon `:`

C. A full stop `.`

D. Nothing

68.1.Q8. Rewrite `x >= 1 and x <= 9` as a single chained comparison.

Answer Key – Chapter 68 – Booleans, Comparisons & Conditionals

68.1.Q1. `True` – `==` tests equality, and 5 does equal 5.

68.1.Q2. `False` – `and` needs both sides true; $4 < 2$ is false, so the whole thing is false.

68.1.Q3. `True` – $10 \neq 10$ is `False`, and `not False` is `True`.

68.1.Q4. A single `=` assigns (e.g. `x = 5`); a double `==` tests equality (e.g. `x == 5`).

68.1.Q5. False – `0`, `""` and `[]` are the everyday falsey values; non-zero numbers are truthy.

68.1.Q6. `medium` – the first true branch is `n > 10`; Python runs it and skips the rest.

68.1.Q7. A colon `:` – every `if`, `elif` and `else` header line ends with a colon, followed by an indented block.

68.1.Q8. `1 <= x <= 9` – Python allows chained comparisons that read just like $1 \leq x \leq 9$.

Chapter 69 · Loops

How to make Python repeat work with for and while loops, generate sequences with range, build up a running total, and put it to work finding factors and testing whether a number is prime.

The Idea Behind It

Why `range(n)` gives exactly n numbers (0 up to $n-1$), with no off-by-one. *but why?*

`range(n)` is **half-open**: it *includes* the start 0 but *stops just before* n. That looks like it might lose a value, yet it is precisely what makes the count come out clean. The reason is a counting fact: the whole numbers from 0 to $n-1$ are $0, 1, 2, \ldots, n-1$, and you count them by the simple rule **last minus first, plus one**:

$$\underbrace{(n-1)}_{\text{last}} - \underbrace{0}_{\text{first}} + 1 = n \quad \text{values}$$

So a half-open range from 0 to n holds exactly n items, and a loop over it runs exactly n times. Starting at 0 rather than 1 is what cancels the $+1$ and kills the classic off-by-one slip. Watch `range(5)` deliver five numbers, ending at 4, not 5:

```
for i in range(5):
    print(i, end=" ")
```

```
output
0 1 2 3 4
```

And a loop that adds a value each pass is just a running **sum** Σ written as instructions: `total += k` for k across the range is the code form of $\sum k$. Summing `range(1, 5)` gives $1+2+3+4$:

$$\sum_{k=1}^{4} k = 1+2+3+4 = 10$$

$$\therefore \texttt{range(n)} \text{ counts } n \text{ items, so the loop runs } n \text{ times}$$

Half-open ranges count cleanly, and a loop is a Σ you can run.

start at 0, stop before n, and the arithmetic just works out

Computers are tireless. The thing they do better than anything else is *repeat* a task – thousands or millions of times – without getting bored or making a slip. A **loop** is how you tell Python to do something again and again. In this chapter we will use loops to do real arithmetic: in particular, to hunt for the factors and primes you met in Chapter 7.

The for loop and range

A **for loop** runs a block of code once for each item in a sequence. The simplest sequence is made by **range**, which produces a run of whole numbers. Like an `if`, the header line ends in a colon and the repeated body is indented.

```
for i in range(5):
    print(i)
```

```
output
0
1
2
3
4
```

Note: `range(5)` counts $0, 1, 2, 3, 4$ – it **starts at** 0 and **stops just before** 5, giving exactly five numbers. This "up to but not including" habit catches everyone at first, so keep it in mind.

You can give `range` a start and a stop, and even a step. `range(1, 6)` counts 1 to 5; `range(2, 11, 2)` counts in twos:

```
for n in range(2, 11, 2):
    print(n)
```

```
output
2
4
6
8
10
```

A loop over `range` is the natural way to print a times table – the multiplication facts from Chapter 4. This shows the 7 times table up to $\times 5$:

```
for i in range(1, 6):
    print("7 x", i, "=", 7 * i)
```

```
output
7 x 1 = 7
7 x 2 = 14
7 x 3 = 21
7 x 4 = 28
7 x 5 = 35
```

Accumulation: building up a total

A wonderfully common pattern is to start a variable at 0 and add to it each time round the loop. This is called **accumulation**. Here we add up the whole numbers from 1 to 10 – the same running total you would do by hand.

```
total = 0
for n in range(1, 11):
    total = total + n
print(total)
```

```
output
55
```

Tip: The line `total = total + n` is so common that Python gives it a shortcut: `total += n` means exactly the same thing. There is also `*=`, `-=` and `/=`.

The same pattern with **multiplication** computes a factorial. Start the running product at 1 and multiply by each number in turn – this works out $5! = 1 \times 2 \times 3 \times 4 \times 5$:

```
product = 1
for n in range(1, 6):
    product *= n
print("5! =", product)
```

```
output
5! = 120
```

The while loop

A **while loop** repeats *as long as* a condition stays true – it is the looping cousin of the `if` statement. Use it when you do not know in advance how many turns you need. Something inside the loop must eventually make the condition false, or it will run forever.

```
count = 3
while count > 0:
    print(count)
    count -= 1
print("Lift off!")
```

```
output
3
2
1
Lift off!
```

Note: If the condition of a `while` loop never becomes false, you get an **infinite loop** that never stops. Always make sure something in the body moves you toward the exit – here `count -= 1` does the job.

Finding the factors of a number

Time to connect this to number theory. Recall from Chapter 7 that a **factor** of n is a number that divides into it with **no remainder** – that is, where $n \bmod d = 0$. The remainder operator % answers exactly that question, so a loop can find every factor of a number:

```
n = 12
for d in range(1, n + 1):
    if n % d == 0:
        print(d)
```

```
output
1
2
3
4
6
12
```

The loop tries every candidate d from 1 up to n. Whenever $12 \bmod d$ is 0, that d divides evenly, so it is a factor. Out come $1, 2, 3, 4, 6, 12$ – exactly the factors of 12.

Testing whether a number is prime

A **prime** number has *no factors* other than 1 and itself. So to test a number, we count how many factors it has. We can reuse our factor idea: start a counter at 0 and add 1 for every divisor we find. A prime ends up with a factor count of exactly 2.

```
n = 13
factors = 0
for d in range(1, n + 1):
    if n % d == 0:
        factors += 1
if factors == 2:
    print("prime")
else:
    print("not prime")
```

```
output
prime
```

13 is divisible only by 1 and 13, so the counter reaches 2 and we declare it prime. Swap in `n = 12` and the counter climbs past 2, printing "not prime". You have just written a real primality test using nothing but a loop and the % operator.

Tip: Once you are comfortable, look up the keyword `break`. It lets a loop stop early – for example, the moment you find a divisor you can declare a number "not prime" without checking the rest.

Quiz

69.1.Q1. Predict the output: `for i in range(3): print(i)`.

69.1.Q2. How many numbers does `range(2, 8)` produce, and what are they?

69.1.Q3. What does `total += 5` do?

A. Sets `total` to `5`

B. Adds `5` to `total`

C. Tests if `total` equals `5`

D. Prints `5`

69.1.Q4. Predict the final value printed: `s = 0 for n in range(1, 5): s += n print(s)`

69.1.Q5. True or False: a `while` loop whose condition never becomes false will stop on its own.

A. True

B. False

69.1.Q6. Which expression is true exactly when `d` is a factor of `n`?

A. `n / d == 0`

B. `n % d == 0`

C. `n // d == 0`

D. `n == d`

69.1.Q7. In the prime test, why do we check whether the factor count equals exactly `2`?

69.1.Q8. Predict the output: `k = 2 while k <= 6: print(k) k += 2`

Answer Key – Chapter 69 – Loops

69.1.Q1. It prints `0`, `1`, `2` on separate lines – `range(3)` starts at 0 and stops before 3.

69.1.Q2. Six numbers: $2, 3, 4, 5, 6, 7$ – it includes the start 2 but excludes the stop 8.

69.1.Q3. Adds `5` to `total` – it is shorthand for `total = total + 5`.

69.1.Q4. `10` – it sums $1 + 2 + 3 + 4 = 10$.

69.1.Q5. False – that is an infinite loop; something in the body must eventually make the condition false.

69.1.Q6. `n % d == 0` – the remainder of $n \div d$ is 0 precisely when d divides n evenly.

69.1.Q7. A prime has exactly two factors – 1 and itself. More than two means it is composite; fewer (like 1) means it is not prime.

69.1.Q8. It prints `2`, `4`, `6` – starting at 2 and adding 2 each pass until k exceeds 6.

Chapter 70 · Functions

How to write your own functions with def, pass them inputs through parameters, hand back a result with return, and reuse them – including coding the function f(x)=3x+2 from Chapter 29.

The Idea Behind It

Why `def f(x): return 3*x + 2` is the function $f(x) = 3x + 2$ from Chapter 29. *but why?*

Chapter 29 defined a function as a rule that takes each input to exactly **one** output, and *calling* it $f(4)$ meant one thing: **substitute** 4 for x everywhere in the rule and simplify. A Python `def` is that same definition, made runnable. The parameter `x` is the placeholder; calling `f(4)` slots 4 into it and `return` hands back the single result.

$$f(x) = 3x + 2 \xrightarrow{x:=4} 3(\,4\,) + 2 = 14$$

Line the two up and they substitute in lockstep, symbol for symbol, the maths and the code reaching the same value because they perform the same substitution:

```
def f(x):
    return 3 * x + 2

print(f(4))
```

```
output
14
```

Because the rule pins each input to one output, the same call always returns the same value, exactly the *single-output* property that makes f a function in the first place (the vertical-line test of Chapter 29).

$$\therefore \texttt{f(4)} = f(4) = 14$$

Calling a function is substitution, the move you already make every time you evaluate f at a number.

def f(x) is f(x), now you can press Enter on it

In Chapter 29 you met the idea of a **function** as a machine: you feed it an input, it follows a rule, and out comes one output. You wrote things like $f(x) = 3x + 2$, then fed in $x = 4$ to get $f(4) = 14$. Python functions are that exact idea made runnable. Once you define a rule, you can use it as many times as you like without rewriting it.

Defining a function with def

You create a function with the keyword **def**, followed by a name, a pair of brackets, and a colon. The indented body is the rule. The word **return** sends a value back out of the machine.

```
def double(n):
    return n * 2

print(double(5))
print(double(20))
```

```
output
10
40
```

Here `n` is a **parameter** – a placeholder for whatever number you pass in. When you write `double(5)`, the value 5 flows into `n`, the rule `n * 2` runs, and `return` hands back 10. The same function happily handles 20 too. Define once, use forever.

Coding f(x) = 3x + 2

Let us translate the algebraic function $f(x) = 3x + 2$ directly into Python. The mathematical notation $f(x)$ and the code `f(x)` line up almost letter for letter:

```
def f(x):
    return 3 * x + 2

print(f(4))    # 3(4) + 2
print(f(0))    # the y-intercept
print(f(-2))
```

```
output
14
2
-4
```

Check it against the maths: $f(4) = 3(4) + 2 = 14$, and $f(0) = 2$ – the y-intercept you would read off the graph. Python and algebra agree, because they are doing the very same arithmetic.

Building a table of values

Remember making tables of values to plot a line? A loop plus a function does it instantly. This prints each x next to its $f(x)$:

```
def f(x):
    return 3 * x + 2

for x in range(0, 4):
    print(x, f(x))
```

```
output
```

```
0 2
1 5
2 8
3 11
```

Each row is an ordered pair $(x, f(x))$ ready to plot: $(0, 2), (1, 5), (2, 8), (3, 11)$. Notice they go up by 3 each step – that steady rise is the **slope** of the line.

Any rule can become a function. Here is `square`, which returns n^2 – the squaring of Chapter 10 – and a loop that prints a small table of squares:

```
def square(n):
    return n ** 2

for n in range(1, 5):
    print(n, square(n))
```

```
output
1 1
2 4
3 9
4 16
```

More than one parameter

A function can take several inputs, separated by commas. Here is a rule with two parameters that computes the area of a rectangle:

```
def area(width, height):
    return width * height

print(area(3, 4))
print(area(10, 2))
```

```
output
12
20
```

Because a function is reusable, you can call it inside a loop to build a whole table of values – exactly the $(x,\ f(x))$ table from Chapter 29:

```
def f(x):
    return 3 * x + 2        # the rule from Chapter 29

for x in range(0, 4):
    print(x, f(x))
```

```
output
0 2
1 5
2 8
3 11
```

A function that answers yes or no

A function can hand back a **boolean** – the `True`/`False` values of Chapter 33 – so it reads like a question. Here `is_even` uses the `%` test from Chapter 32 and gives a clean yes/no answer:

```python
def is_even(n):
    return n % 2 == 0

print(is_even(10))
print(is_even(7))
```

```
output
True
False
```

Giving the rule a name like `is_even` makes the code that *uses* it read almost like English: `if is_even(x):`. The function packages the idea once, and every reader instantly knows what it asks.

Why functions are worth it

Functions let you give a name to an idea and reuse it. Write a tricky calculation **once**, get it right **once**, and then call it by name wherever you need it. Your programs become shorter, clearer, and far easier to fix.

Note: A function stops the moment it hits a `return`, handing that value back to wherever it was called. A function with no `return` still runs its body, but it hands back the special value `None` – the Python way of saying "nothing to report".

Tip: Choose names that say what the function does: `area`, `is_prime`, `to_percent`. Future-you, reading the code next month, will be grateful for a name that explains itself.

Quiz

70.1.Q1. Which keyword begins the definition of a function in Python?

A. `func`

B. `def`

C. `function`

D. `define`

70.1.Q2. In `def double(n): return n * 2`, what is `n` called?

A. A return value

B. A parameter

C. A keyword

D. An output

70.1.Q3. Given `def f(x): return 3 * x + 2`, predict `print(f(5))`.

70.1.Q4. What does the `return` keyword do?

70.1.Q5. Predict the output: `def f(x): return 3 * x + 2 print(f(0))`

70.1.Q6. True or False: once you define a function, you may call it as many times as you like.

A. True

B. False

70.1.Q7. Write a function `square` that returns the square of its input, then state `square(6)`.

70.1.Q8. How many parameters does `def area(width, height): return width * height` take, and what does `area(5, 2)` return?

Answer Key – Chapter 70 – Functions

70.1.Q1. `def` – for example, `def greet():`.

70.1.Q2. A parameter – a placeholder that receives the value you pass in when you call the function.

70.1.Q3. 17 – $3(5) + 2 = 17$.

70.1.Q4. It sends a value back out of the function to wherever it was called, and the function stops at that point.

70.1.Q5. 2 – $3(0) + 2 = 2$, which is the y-intercept of the line $f(x) = 3x + 2$.

70.1.Q6. True – defining it once lets you reuse the rule everywhere, with different inputs.

70.1.Q7. `def square(n): return n * n` gives `square(6)` = 36.

70.1.Q8. Two parameters; `area(5, 2)` returns `10` (5×2).

Chapter 71 · Collections: Lists, Sets & Dictionaries

How to group many values together using lists, sets and dictionaries – reading items by index, testing membership with in, looping over them, and seeing how Python sets mirror the union and intersection from Chapter 23.

The Idea Behind It

Why a Python `set` is a mathematical set: $\{1, 2\} = \{2, 1, 1\}$. *but why?*

A set in Chapter 23 was defined by one question only: *is this element in or out?* Nothing else counts, so a set has **no order** (no first or last) and **no duplicates** (an element is in once or not at all). Listing 1 twice, or in a different order, describes the very same set, because the in-or-out answers are identical.

$$\{1, 2\} \;=\; \{2, 1\} \;=\; \{2, 1, 1\} \quad (\text{same members, so the same set})$$

Python's `set` type carries that definition over exactly: it stores each value at most once and keeps no positions, so it reports those three spellings as equal and silently drops the repeat:

```
print({1, 2} == {2, 1, 1})
print({2, 1, 1})
```

```
output
True
{1, 2}
```

And because the objects match, so do the operations: union $\cup$, intersection $\cap$ and difference $\setminus$ from Part V are built in as `|`, `&` and `-`, obeying the same Venn-diagram laws.

$$A \cup B, \quad A \cap B, \quad A \setminus B \;\longleftrightarrow\; \texttt{A | B, A \& B, A - B}$$

$$\therefore \text{ the data type } \textit{is} \text{ the math object, not just a copy of it}$$

So everything you proved about sets on paper holds for sets in code, untouched.

membership is all that matters, in maths and in Python alike

So far each variable has held a single value. But real problems come in bunches: a class of test scores, the days of the week, the prices in a shop. Python gives you **collections** – single objects that hold many values at once. Three are essential: the **list**, the **set**, and the **dictionary**. You have already met the mathematics behind sets in Chapter 23, so you are halfway there.

Lists: an ordered collection

A **list** is an ordered row of values, written inside square brackets and separated by commas. The order is kept, and values may repeat.

```
scores = [88, 72, 95, 60]
print(scores)
print(len(scores))   # how many items?
```

```
output
[88, 72, 95, 60]
4
```

Reading items by index

Each item has a position, called its **index**. Crucially, indexing **starts at** 0 – the first item is index 0, the second is index 1, and so on. This is the same "count from zero" habit you saw with `range`.

```
scores = [88, 72, 95, 60]
print(scores[0])    # first item
print(scores[2])    # third item
print(scores[-1])   # last item
```

```
output
88
95
60
```

Tip: A negative index counts from the back: `scores[-1]` is the last item, `scores[-2]` the second-to-last. Handy when you do not know the length.

Membership and iteration

The keyword **in** asks whether a value is present – and it returns a boolean, linking straight back to your logic chapters. A `for` loop walks through every item in turn.

```
scores = [88, 72, 95, 60]
print(95 in scores)
print(100 in scores)
for s in scores:
    print(s)
```

```
output
True
False
88
72
95
60
```

Looping over a list pairs naturally with the accumulation pattern from Chapter 34. Here we total the scores and divide by how many there are to get the average:

```
scores = [88, 72, 95, 65]
total = 0
for s in scores:
    total += s
print("average:", total / len(scores))
```

```
output
average: 80.0
```

Note: Python also has built-ins for this: <code>sum(scores)</code> adds a list for you and <code>max(scores)</code> finds the largest, so <code>sum([88, 72, 95, 65]) / 4</code> gives the same <code>80.0</code>.

Sets: a collection with no duplicates

A **set** is Python's version of the mathematical set from Chapter 23: an *unordered* collection with **no repeats**. Write it with curly braces. Adding a value that is already there changes nothing.

```
a = {1, 2, 3, 4}
b = {3, 4, 5, 6}
print(a | b)    # union  (A ∪ B)
print(a & b)    # intersection  (A ∩ B)
print(a - b)    # difference  (A − B)
```

```
output
{1, 2, 3, 4, 5, 6}
{3, 4}
{1, 2}
```

These are exactly the operations you drew with Venn diagrams. The **union** `|` gathers everything in either set; the **intersection** `&` keeps only what is in **both**; the **difference** `-` takes what is in the first but not the second. The OR / AND link from logic is right there in the symbols.

Note: Because a set throws away duplicates, wrapping a list in `set(...)` is a quick way to remove repeats: `set([1, 2, 2, 3])` gives `{1, 2, 3}`.

That "no repeats" rule has a neat use: feed the letters of a word into a set and you are left with only the **distinct** letters – the kind of counting question sets were built for in Chapter 23. A set has no fixed order, so we `sorted(...)` it for a tidy display:

```
word = "banana"
letters = set(word)
print(sorted(letters))
print(len(letters), "distinct letters")
```

```
output
['a', 'b', 'n']
3 distinct letters
```

Dictionaries: labelled values

A **dictionary** stores **key–value pairs**: instead of looking things up by position, you look them up by a meaningful label. Think of a real dictionary – a *word* (the key) points to its *meaning* (the value).

```
ages = {"Ada": 36, "Alan": 41}
print(ages["Ada"])      # look up by key
ages["Grace"] = 45      # add a new pair
print(ages)
```

```
output
36
{'Ada': 36, 'Alan': 41, 'Grace': 45}
```

You retrieve a value by its key in square brackets, and you add a new pair just by assigning to a fresh key. The keyword `in` works here too, testing whether a **key** is present:

```
ages = {"Ada": 36, "Alan": 41}
print("Ada" in ages)
print("Bob" in ages)
```

```
output
True
False
```

Which one should I use?

A quick guide:

- Use a **list** when order matters or values may repeat (scores in the order taken, steps in a recipe).
- Use a **set** when you only care *which* values are present, with no duplicates (the distinct letters in a word).
- Use a **dictionary** when each value needs a label to find it by (a name to an age, a product to its price).

Quiz

71.1.Q1. What does `len([4, 8, 15, 16])` return?

71.1.Q2. Given `xs = [10, 20, 30]`, what is `xs[0]`?

A. `10`

B. `20`

C. `30`

D. an error

71.1.Q3. Given `xs = [10, 20, 30]`, predict `print(xs[-1])`.

71.1.Q4. Predict the output: `print(5 in [1, 3, 5, 7])`.

71.1.Q5. For `a = {1, 2, 3}` and `b = {2, 3, 4}`, what is `a & b`?

A. `{1, 2, 3, 4}`

B. `{2, 3}`

C. `{1, 4}`

D. `{}`

71.1.Q6. For the same `a` and `b`, what does `a | b` give, and which set operation is it?

71.1.Q7. True or False: a Python set can contain the same value twice.

A. True

B. False

71.1.Q8. Given `d = {"x": 1, "y": 2}`, how do you read the value stored under the key `"y"`?

Answer Key – Chapter 71 – Collections: Lists, Sets & Dictionaries

71.1.Q1. `4` – `len` gives the number of items in the list.

71.1.Q2. `10` – list indexing starts at 0, so index 0 is the first item.

71.1.Q3. `30` – a negative index counts from the end, so -1 is the last item.

71.1.Q4. `True` – `in` checks membership, and 5 is in the list.

71.1.Q5. `{2, 3}` – `&` is intersection, the values in **both** sets, matching $A \cap B$.

71.1.Q6. `{1, 2, 3, 4}` – `|` is union ($A \cup B$), everything in either set, with no duplicates.

71.1.Q7. False – sets discard duplicates, just like sets in mathematics.

71.1.Q8. Write `d["y"]`, which returns `2` – you look up a dictionary value by its key in square brackets.

Chapter 72 · Putting It Together: Small Programs

How to combine variables, conditionals, loops and functions into complete little programs that solve problems from earlier in the book – a percentage calculator, a prime checker, and an equation-solution checker.

The Idea Behind It

Why splitting a program into small functions is the way you already solve hard maths. *but why?*

You have done this all book. To work out $\frac{2^5 \times 2^4}{2^3}$ you did not attack it whole: you solved the top, solved the bottom, then combined. A multi-step proof is the same, a chain of small claims each true on its own. The move is **decomposition**: break a hard problem into sub-problems you can solve *and check* separately, then compose the pieces.

$$\frac{2^5 \times 2^4}{2^3} = \frac{2^9}{2^3} = 2^6 \quad \text{(small steps, each checkable)}$$

A program is built the very same way. Each `def` names one sub-problem, gets it right once, and can be tested in isolation; the main program just composes those trusted parts. Here a hard task, "is n prime?", is split into a clean factor test you can check on its own, then reused to list primes:

```
def is_prime(n):
    if n < 2:
        return False
    for d in range(2, n):
        if n % d == 0:
            return False
    return True

print(is_prime(7), is_prime(8))
```

```
output
True False
```

Because each function is correct alone, the whole is correct by composition, exactly the reasoning that lets a proof's lemmas combine into a theorem.

$$\therefore \text{ solve the parts, check the parts, compose the whole}$$

Decomposition is one habit wearing two hats: the mathematician's and the programmer's.

break it down, get each piece right, snap them together

You now have every building block: values and variables, the comparisons and `if` statements of Chapter 33, the loops of Chapter 34, the functions of Chapter 35, and the collections of Chapter 36. A **program** is simply these pieces assembled to get a job done. In this chapter we write three small but complete programs – and each one solves a problem you have already met by hand earlier in the book. Coding is just maths you can run.

Program 1 – A percentage calculator

Back in Chapter 16 you worked out "what is 20% of 80?" The recipe was: turn the percent into a decimal, then multiply. Let us capture that as a reusable function.

```
def percent_of(percent, whole):
    return percent / 100 * whole

print(percent_of(20, 80))    # 20% of 80
print(percent_of(15, 200))   # 15% of 200
```

```
output
16.0
30.0
```

The function does exactly what you did with pencil: $20 \div 100 \times 80 = 16$. Now any percentage problem is one short call away. We can even format the answer nicely – an **f-string** (a string with an `f` in front) lets us drop a value straight into text:

```
def percent_of(percent, whole):
    return percent / 100 * whole

result = percent_of(15, 200)
print(f"15% of 200 is {result}")
```

```
output
15% of 200 is 30.0
```

Program 2 – A prime checker

In Chapter 7 you tested numbers for primality by hunting for factors. Let us package that into a clean function called `is_prime` that returns a boolean – `True` or `False` – so it reads like a yes/no question.

```
def is_prime(n):
    if n < 2:
        return False
    for d in range(2, n):
        if n % d == 0:
            return False
    return True

print(is_prime(13))
print(is_prime(12))
```

```
output
```

```
True
False
```

Read it as plain logic. Anything below 2 is not prime. Otherwise we try each divisor d from 2 up to $n-1$: if **any** divides evenly ($n \bmod d = 0$), we `return False` at once – no need to look further. If the loop finishes finding none, the number is prime, so we `return True`.

Because `is_prime` hands back a boolean, we can drop it straight into a loop to list the primes up to 20 – exactly the kind of table you might have built by hand:

```python
def is_prime(n):
    if n < 2:
        return False
    for d in range(2, n):
        if n % d == 0:
            return False
    return True

for n in range(2, 21):
    if is_prime(n):
        print(n)
```

```
output
2
3
5
7
11
13
17
19
```

Program 3 – An equation-solution checker

In Chapter 27 you solved equations and then **checked** your answer by substituting it back in. That checking step is something Python does perfectly. Suppose we claim $x = 5$ solves $3x + 2 = 17$. We substitute and compare the two sides with ==:

```python
x = 5
left = 3 * x + 2
right = 17
print(left == right)
```

```
output
True
```

The left side comes to $3(5) + 2 = 17$, the right side is 17, and == confirms they match: `True`. Wrap it in a function and you have a reusable checker for any candidate solution:

```
def checks_out(x):
    return 3 * x + 2 == 17

print(checks_out(5))
print(checks_out(4))
```

```
output
True
False
```

Feeding in 5 gives `True` (it is the solution); feeding in 4 gives `False` ($3(4) + 2 = 14 \neq 17$). The computer has just verified your algebra.

Program 4 – A temperature-conversion table

A function and a loop together build a whole table at once. This codes the rule $F = \frac{9}{5}C + 32$ from Chapter 35 as a function, then prints Celsius beside Fahrenheit for $0, 10, 20, 30$ degrees:

```
def to_fahrenheit(c):
    return 9 / 5 * c + 32

for c in range(0, 40, 10):
    print(c, "C  =", to_fahrenheit(c), "F")
```

```
output
0 C  = 32.0 F
10 C  = 50.0 F
20 C  = 68.0 F
30 C  = 86.0 F
```

Each row is an ordered pair (C, F), just like the tables of values you plotted in Chapter 29 – only now Python fills the whole table in one short loop.

Program 5 – A greatest-common-factor finder

In Chapter 8 you found the **greatest common factor** of two numbers by listing what divides both. A loop does the listing for you: try every candidate up to the smaller number and keep the largest that divides **both** with no remainder (the % test again).

```python
def gcf(a, b):
    best = 1
    for d in range(1, min(a, b) + 1):
        if a % d == 0 and b % d == 0:
            best = d
    return best

print(gcf(12, 18))
```

```
output
6
```

The loop checks $1, 2, \ldots$ up to 12; the divisors common to both 12 and 18 are $1, 2, 3, 6$, and the last one kept – the largest – is 6. That is exactly the GCF you would find by hand.

What makes these programs

Notice the shared shape. Each program names a clear idea with a **function**, uses **conditionals** and sometimes a **loop** inside, and hands back a **result** you can print or reuse. That is the whole craft in miniature: break a problem into small rules, write each as a function, and combine them.

Note: Every program here started life as a problem you could already solve on paper. That is the real lesson of this book: once you understand the maths and the logic, turning it into code is mostly a matter of writing the steps in the grammar of Python.

Tip: Try changing the numbers in each program and predicting the output before you run it. When the result matches your prediction, you truly understand the code.

Quiz

72.1.Q1. In `percent_of`, why do we divide the percent by `100`?

72.1.Q2. Predict the output: `print(percent_of(50, 90))` using `def percent_of(p, w): return p / 100 * w`.

72.1.Q3. In `is_prime`, what does the function return for `n = 1`?

A. `True`

B. `False`

C. `1`

D. an error

72.1.Q4. Why does `is_prime` use `return False` as soon as it finds a divisor, instead of finishing the loop?

72.1.Q5. True or False: `is_prime(13)` returns `True`.

A. True

B. False

72.1.Q6. In the equation checker, what is the difference between the single = and the double == being used?

72.1.Q7. Predict the output: `print(checks_out(4))` where `def checks_out(x): return 3 * x + 2 == 17`.

72.1.Q8. What value would make `checks_out(x)` return `True`, and why?

Answer Key – Chapter 72 – Putting It Together: Small Programs

72.1.Q1. To turn a percent into a decimal first – e.g. 20% becomes 0.2 – then multiply by the whole, exactly as in Chapter 16.

72.1.Q2. `45.0` – 50% of 90 is $0.5 \times 90 = 45$.

72.1.Q3. `False` – the first check `if n < 2: return False` rules out 0 and 1, which are not prime.

72.1.Q4. Finding even one divisor proves the number is composite, so there is no reason to keep checking – returning early is correct and faster.

72.1.Q5. True – 13 has no divisor between 2 and 12, so the loop finishes and returns `True`.

72.1.Q6. A single = assigns (e.g. `x = 5`); the double == *tests* whether the two sides of the equation are equal.

72.1.Q7. `False` – $3(4) + 2 = 14$, which is not 17, so the comparison is false.

72.1.Q8. `x = 5`, because $3(5) + 2 = 17$ matches the right-hand side – it is the solution of $3x + 2 = 17$.

Chapter 73 · Where to Go Next

> *How to keep your momentum going: installing Python on your own computer, choosing a friendly editor, project ideas to practise on, and what to learn next.*

The Idea Behind It

You already know the hard part: the math and logic of this book are the foundations of programming.

Look back at what these Python chapters actually asked of you. A *variable* is the x of algebra, a labelled value you reuse. A *function* is $f(x)$, one input mapped to one output. *Booleans* with `and`/`or`/`not` are the AND, OR, NOT of Part V. A *loop* that sums is the Σ you already write. A *set* is the set of Chapter 23, union and intersection included. None of it was a new concept, it was a new **notation** for ideas that were already yours.

So learning a programming language is mostly learning where the brackets and colons go, not learning to think in a new way. The thinking, decomposing a problem, reasoning in true and false, evaluating a rule at an input, you have been practising for this whole book.

That is why the next steps feel less like climbing and more like exploring. A second language (JavaScript, say, or C) reuses these same foundations under different punctuation. **Keep going:** you are not starting from zero, you are building on a foundation you have already poured.

the concepts are yours; from here it is just vocabulary

Congratulations – you have come a remarkably long way. You started with the order of operations and you can now write functions, loops and complete little programs. Most importantly, you have seen the secret hiding in plain sight: **programming is the maths and logic you already understand, written in a language a computer can run.** This final chapter is your map for what comes next. There is no quiz to fear here, just encouragement and a few signposts.

Step 1: Put Python on your own computer

Everything so far runs in Python, the most beginner-friendly major language there is. To run it yourself:

1. Go to the official website, **python.org**, and click *Downloads*. It will offer the right version for your computer automatically.
2. Run the installer. On Windows, **tick the box that says "Add Python to PATH"** – this one checkbox saves a lot of headaches later.
3. To check it worked, open a terminal (Command Prompt on Windows, Terminal on a Mac) and type `python --version`. If it prints a version number, you are ready.

That terminal is also a live **REPL**: type `python`, press Enter, and you can try expressions one line at a time, exactly like the very first examples in this part.

```
>>> 3 + 4 * 2
11
>>> "hello" * 3
'hellohellohello'
```

```
output
(the >>> is Python's prompt — you type after it)
```

Step 2: Choose a friendly editor

The REPL is great for quick experiments, but real programs live in files. A **code editor** gives you colour-coding, helpful suggestions, and a button to run your file. Two excellent, free choices:

- **Visual Studio Code** (often just "VS Code") – the most popular editor in the world. Install it, add the official **Python extension**, and you can write a file like `hello.py` and run it with a single click.
- **Thonny** – built specifically for Python beginners. It bundles Python *and* an editor together in one small download, so there is nothing else to set up. A lovely place to start.

> **Tip:** Save your programs in files ending in `.py`. To run a file from the terminal, type `python hello.py`. Give files clear names – `prime_checker.py` beats `stuff.py` every time.

Step 3: Practise with small projects

You learn to code by *coding*. The best next step is to build tiny things that interest you. Each of these uses only what this book has taught:

- **A times-tables printer** – a loop that prints the multiplication table for any number (Chapter 4 and Chapter 34).
- **A factor finder** – ask for a number and list all its factors (Chapter 7).
- **A grade calculator** – read a score and print the grade with `if/elif/else` (Chapter 33).
- **A temperature converter** – a function that turns Celsius into Fahrenheit using $F = \frac{9}{5}C + 32$ (Chapter 35).
- **A tip calculator** – work out a percentage of a bill and the new total (Chapter 16 and Chapter 37).

To make programs that talk back to you, learn the **input** function – it pauses and waits for the user to type something:

```
name = input("What is your name? ")
print("Hello, " + name + "!")
```

```
output
What is your name? Ada
Hello, Ada!
```

Put a few ideas together and you have a genuinely useful program. Here is a complete tip calculator that asks for a bill and prints 15% of it plus the new total (the percentages of Chapter 16):

```
bill = float(input("Bill amount: "))
tip = bill * 0.15            # 15%, from Chapter 16
print("Tip:", tip)
print("Total:", bill + tip)
```

```
output
Bill amount: 40
Tip: 6.0
Total: 46.0
```

Step 4: What to learn next

When you are ready for more, here is a natural path – each item builds gently on this book:

- **Strings in depth** – slicing text, searching, and useful methods like `.upper()` and `.split()`.
- **Reading and writing files** – so your programs can save their results and remember things.
- **The standard library** – ready-made tools that come with Python, such as `math` (square roots, π) and `random` (dice, shuffling).
- **Error handling** – `try` and `except`, so a program copes gracefully when something goes wrong.
- **A bigger project** – a quiz game, a simple calculator app, or a number-guessing game that ties everything together.

Two of those tools are worth a peek right now. The **`math`** library hands you the square roots and π of Chapters 11 and 47:

```
import math

print(math.sqrt(144))        # square root, Chapter 11
print(round(math.pi, 4))     # pi, Chapter 47
```

```
output
12.0
3.1416
```

With `math.pi` in hand you can code the area of a circle, $A = \pi r^2$, straight from Chapter 47. Here is the area of a circle of radius 5, rounded to two decimals:

```
import math

radius = 5
area = math.pi * radius ** 2
print(round(area, 2))
```

```
output
```

```
78.54
```

And the **`random`** library powers games. Here is the heart of a number-guessing game that ties together variables, loops and the conditionals of Part V:

```python
import random

secret = random.randint(1, 10)   # a hidden number 1–10
guess = int(input("Guess 1–10: "))
if guess == secret:
    print("Correct!")
else:
    print("Nope, it was", secret)
```

```
output
Guess 1–10: 7
Nope, it was 4
```

The same `random` tool rolls dice. A loop that rolls a die a few times mixes everything you have learned – a loop, a function call, and a list to collect the results:

```python
import random

rolls = []
for i in range(5):
    rolls.append(random.randint(1, 6))
print(rolls)
```

```
output
[4, 1, 6, 3, 2]
```

Each `random.randint(1, 6)` is one roll of a fair six-sided die, and `append` adds it to the list `rolls` – run it again and you will get a different five numbers, just like real dice.

A warm closing

Every professional programmer began exactly where you are now: with the basics, a willingness to try, and the patience to fix what breaks. You already have the two things that matter most – the **logical thinking** from Part V and the **mathematical fluency** from the rest of this book. The syntax will come quickly because the ideas underneath it are already yours.

So open an editor, type a few lines, and run them. Make mistakes – they are how everyone learns. Be curious, be patient, and above all, **keep going.** You are a programmer now. Welcome.

> **Note:** Whenever you get stuck, remember that the official Python documentation at **docs.python.org** and a quick search engine query are part of every programmer's daily toolkit. Looking things up is not cheating – it is the job.

Quiz

73.1.Q1. Where do you download the official version of Python?

A. `python.org`

B. `google.com`

C. the App Store only

D. it comes pre-built in every browser

73.1.Q2. On Windows, which checkbox in the installer is important to tick?

73.1.Q3. Name one beginner-friendly code editor mentioned in this chapter.

73.1.Q4. What file extension should you give your Python programs?

A. `.py`

B. `.txt`

C. `.python`

D. `.exe`

73.1.Q5. What does the `input()` function do?

73.1.Q6. True or False: looking things up in the documentation or online is a normal part of programming.

A. True

B. False

Answer Key – Chapter 73 – Where to Go Next

73.1.Q1. `python.org` – the official site offers the correct installer for your computer.

73.1.Q2. "Add Python to PATH" – it lets you run `python` from any terminal without extra setup.

73.1.Q3. Visual Studio Code (with the Python extension) or Thonny – both are free and excellent for starting out.

73.1.Q4. `.py` – for example, `prime_checker.py`. You run it with `python prime_checker.py`.

73.1.Q5. It pauses the program and waits for the user to type something, then hands back what they typed as text.

73.1.Q6. True – every programmer searches and reads docs daily; it is part of the job, not cheating.

Appendix A – Mental Math Toolkit

The fast techniques from across the book, gathered for quick reference. Each is explained in full where it first appears.

Appendix B – Glossary

AABB. an axis-aligned bounding box: an upright rectangle wrapping an object

absolute value. the distance of a number from zero, written $|a|$; always ≥ 0.

addend. a number being added; in $a + b$, both a and b are addends.

adjacent. The side next to the angle you are working with that is not the hypotenuse.

algebraic expression. A combination of numbers, variables, and operations with no equals sign.

amplitude. half the height of a wave from peak to trough; the value $|A|$ in $y = A\sin(Bx + C)$

angle. the amount of turn between two rays meeting at a vertex, measured in degrees

arc. A connected piece of the edge (circumference) of a circle.

arccosine. the inverse of cosine: $\arccos x$ is the angle whose cosine is x, with output in $[0^\circ, 180^\circ]$

arcsine. the inverse of sine: $\arcsin x$ is the angle whose sine is x, with output in $[-90^\circ, 90^\circ]$

arctangent. the inverse of tangent: $\arctan x$ is the angle whose tangent is x, with output in $(-90^\circ, 90^\circ)$

area. the amount of flat surface a shape covers; in square units, m²

arithmetic sequence. a sequence where each term is the previous one plus a fixed common difference d

ASCII. an early code mapping characters to numbers, for example A is 65.

Average rate. The total change divided by the total time; the slope of the secant line between two points.

base. the number of distinct digits a system uses, and the value each place is multiplied by; also called the radix.

biconditional. The "if and only if" statement $p \Leftrightarrow q$; true exactly when p and q have the same truth value.

binary. base two; the number system of 0 and 1 that computers use.

bit. a single binary digit, 0 or 1; the smallest unit of information.

Boolean variable. a quantity that is either 1 (true) or 0 (false)

borrow. to take one unit from the column to the left (regrouping) when subtracting a larger digit from a smaller one.

bounding circle. a circle drawn around an object, used for fast distance-based collision tests

byte. a group of 8 bits; holds 256 values (0 to 255), enough for one text character.

carry. to move a ten (or hundred…) into the next column left when a column total reaches ten.

chord. a straight line joining two points on a circle (not through the centre in general)

circumference. the distance once around the outside of a circle

coefficient. the number multiplying a power of the variable in a term

collinear. lying on one common straight line

collision. the condition that two shapes touch or overlap

common denominator. A shared multiple of two denominators, used to add or subtract fractions; the least one is the LCM.

commutative. an operation where order does not change the result: $a \times b = b \times a$.

complement. the set $\complement A$ of elements of the universe **not** in A

complementary. two angles that add to 90°

component. one of the numbers inside a vector – e.g. the x- or y-part

composite number. A whole number greater than 1 that has a factor other than 1 and itself.

compound inequality. two inequalities joined into one statement, e.g. $-1 \leq x < 4$

compound statement. A statement built from simpler ones using connectives.

conditional. The "if p then q" statement $p \Rightarrow q$; false only when p is true and q is false.

congruent. identical in size and shape (equal sides and angles)

conjunction. The "and" statement $p \wedge q$, true only when both p and q are true.

connective. A word/symbol ($\wedge$, $\vee$, $\neg$, . . .) that combines or modifies statements to form a new one.

constant. A fixed number whose value does not change.

Continuous. A function with no breaks, holes, or jumps; one you can trace without lifting your pen.

contradiction. A statement whose truth-table column is all F – false under every assignment.

contrapositive. The statement $\neg q \Rightarrow \neg p$; always logically equivalent to the original conditional $p \Rightarrow q$.

converse. The statement $q \Rightarrow p$ obtained by swapping the parts of $p \Rightarrow q$; NOT equivalent to the original.

coordinate plane. the grid formed by the horizontal x-axis and vertical y-axis

counterexample. a single value that makes a predicate false, thereby disproving a "for all" claim

Critical point. A point where the derivative is zero, so the tangent is flat; the candidate locations for a maximum or minimum.

De Morgan's laws. The rules $\neg(p \wedge q) \equiv \neg p \vee \neg q$ and $\neg(p \vee q) \equiv \neg p \wedge \neg q$: negate each part and swap the connective.

decimal. base ten; the everyday number system using digits 0 to 9.

decimal point. the dot separating whole-number columns from fractional ones (tenths, hundredths).

degree. the number of edges meeting at a node – how many connections it has

denominator. The bottom number of a fraction; how many equal parts the whole is divided into.

diameter. a straight line across a circle through the centre; twice the radius

difference. the result of a subtraction.

digit. a single symbol in a number; base b has digits 0 to $b - 1$.

disjunction. The (inclusive) "or" statement $p \vee q$, true when at least one of p, q is true.

distance formula. the straight-line distance between two points, from Pythagoras

distributive law. The equivalences $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$ and $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$.

distributive property. The rule $a(b + c) = ab + ac$: a factor outside a bracket multiplies each term inside.

dividend. the number being divided (the "whole" you are splitting).

divisible. one number is divisible by another when the division leaves remainder 0.

divisor. the number you divide by (the size of each group, or the number of groups).

domain. the set of all inputs of a function (the left column)

double negation. The equivalence $\neg(\neg p) \equiv p$: two negations cancel.

duality. every Boolean law has a partner found by swapping $\cdot \leftrightarrow +$ and $0 \leftrightarrow 1$

edge. a connection joining two nodes

encoding. an agreed code book saying which bit-pattern represents which thing.

equation. A statement that two expressions are equal, containing an $=$ sign and (usually) an unknown to solve for.

equivalent fractions. Different-looking fractions that name the same value, e.g. $\frac{1}{2} = \frac{2}{4} = \frac{3}{6}$.

exclusive or. The "one or the other but not both" sense ($\oplus$); the common everyday meaning, but NOT the default in logic.

existential quantifier. $\exists$, "there exists" – claims a predicate holds for *at least one* element

expand. To remove a bracket by distributing, turning $a(b + c)$ into $ab + ac$.

expanded form. a number written as the sum of each digit times its place value, e.g. $304 = 300 + 4$.

exponent. the small raised number that says how many times the base is used as a factor

exponential. an expression b^x with the variable in the exponent; grows by multiplying by b each step

expression. A combination of numbers, variables and operations with no $=$ sign; you simplify or evaluate it, you do not solve it.

factor. A whole number that divides another exactly, with no remainder.

factor (verb). To pull a common factor out front, turning $ab + ac$ into $a(b + c)$ – the reverse of expanding.

factor-label method. converting by multiplying with unit fractions chosen so unwanted units cancel; also called dimensional analysis

fallacy. an argument whose form does not guarantee its conclusion

file format. the rules, signalled by the extension and header, for interpreting the bytes in a file.

function. a relation where every input is paired with exactly one output

geometric sequence. a sequence where each term is the previous one times a fixed common ratio r

gradient. the steepness of a line, measured as rise over run

greatest common factor. The largest number that divides two (or more) numbers exactly; also called HCF or GCD.

hexadecimal. base 16; uses 0 to 9 then A to F for 10 to 15.

hypotenuse. the longest side of a right triangle, opposite the right angle

hypothesis. The "if" part p of a conditional $p \Rightarrow q$ (also called the antecedent).

idempotent law. $A \cdot A = A$ and $A + A = A$: combining a value with itself changes nothing

identity. An equation that holds for every allowed value of its variable (contrast with an equation that is true only for specific values).

image. the shape after a transformation; its points are written with a prime, e.g. A'

improper fraction. A fraction whose numerator is greater than or equal to its denominator, so its value is at least 1.

inclusive or. The logical $\vee$: "p or q or both." The default meaning of "or" in mathematics.

inequality. a statement comparing two quantities with $<$, $>$, $\leq$ or $\geq$ rather than $=$

Instantaneous rate. The rate at a single instant; the slope of the tangent line, equal to the derivative at that point.

integer. a whole number or its negative: $\ldots, -2, -1, 0, 1, 2, \ldots$ (the set $\mathbb{Z}$).

intersection. the set $A \cap B$ of elements in **both** A and B

inverse. an operation that undoes another; logarithm and exponential undo each other

inverse operation. An operation that undoes another: $+$ and $-$ are inverses, as are $\times$ and $\div$.

isometric view. a 3-D-looking drawing where the three axes are drawn at equal angles

least common multiple. The smallest positive number that is a multiple of two (or more) numbers.

like terms. Terms with identical variable parts, which can be combined by adding coefficients.

Limit. The value a function approaches as its input approaches a chosen point.

line. a straight path with no thickness that runs forever in both directions

line segment. the part of a line between two endpoints, with a definite length

linear equation. An equation in which the variable appears only to the first power, so its graph is a straight line.

linear function. a function of the form $f(x) = mx + b$ whose graph is a straight line

logarithm. the exponent needed on a base to produce a given number: $\log_b y = x$ means $b^x = y$

logically equivalent. Two statements ($A \equiv B$) whose truth-table columns are identical for every assignment; freely interchangeable.

lowest terms. A fraction with numerator and denominator divided by their GCF, so they share no factor but 1.

magnitude. the length of a vector, $\sqrt{x^2 + y^2}$

matrix. a rectangular grid of numbers arranged in rows and columns

Maximum. The highest value a function reaches (a peak), where the curve stops rising and starts falling.

midpoint. the point halfway between two points; the average of their coordinates

Minimum. The lowest value a function reaches (a valley), where the curve stops falling and starts rising.

modulus. the fixed number m you divide by and wrap around; the size of the cycle

modus ponens. from $p \Rightarrow q$ and p, infer q

modus tollens. from $p \Rightarrow q$ and $\neg q$, infer $\neg p$

multiple. A number obtained by multiplying a given number by a whole number; the reverse relationship to a factor.

multiplier. The single factor that applies a percentage change: $1 + \frac{r}{100}$ for a rise, $1 - \frac{r}{100}$ for a fall.

negation. The statement $\neg p$ ("not p") that is true exactly when p is false.

negative number. a number less than zero, written with a leading "$-$".

node. a single item in a graph (a person, place, computer); also called a vertex

number line. a line on which numbers are marked in order, smaller to the left, larger to the right.

numerator. The top number of a fraction; how many equal parts are taken.

One-sided limit. The value a function approaches as the input nears a point from just one side (left or right).

open sentence. A sentence containing an unknown (such as $x > 5$) whose truth value is not fixed until the unknown is specified.

opposite. The side directly across from the angle you are working with.

order of magnitude. the nearest power of ten to a quantity, used for rough comparison.

ordered pair. two items written in a fixed order, (a, b), where position matters

orthographic view. a flat, straight-on view of one face, with no perspective distortion

partial product. one piece of a multi-digit product, found by multiplying by a single place value before adding.

path. a route from one node to another, hopping along edges

percentage. A fraction expressed out of 100; $n\% = \frac{n}{100}$.

percentage increase. Adding a percentage of a quantity to itself, multiplying by a factor greater than 1.

perfect square. a whole number that is the square of an integer, e.g. 1, 4, 9, 16, 25

perimeter. the total distance around the outside of a flat shape; a length, in m

period. the horizontal length of one complete repeat; $2\pi/|B|$ for $\sin(Bx)$

phase shift. a horizontal slide of the whole wave, equal to $-C/B$

place value. the value of a column, a power of the base, used to weight its digit.

place-holder. a 0 used to keep other digits in their correct columns, as in 507.

plane. a perfectly flat surface that extends forever in every direction

point. an exact location with no size, labelled by a capital letter

polar coordinates. a point given as (r, θ): distance r from the origin and direction angle θ

pole. the origin, the fixed point from which polar distance is measured

polynomial. a sum of terms, each a coefficient times a whole-number power of the variable

power of ten. one of $10, 100, 1000, \ldots$; multiplying by these just shifts digits left.

predicate. a sentence containing a variable, e.g. "$x > 0$", that becomes true or false once the variable is fixed

prime factorization. A number written as a product of primes; by the Fundamental Theorem of Arithmetic it is unique.

prime number. A whole number greater than 1 whose only factors are 1 and itself.

principal. The original sum of money on which interest is calculated.

product. the result of a multiplication.

projection. representing something in fewer dimensions by dropping or combining coordinates

proposition. Another word for a statement: a sentence with a definite truth value.

Pythagorean identity. The identity cos squared theta + sin squared theta = 1, true for every angle theta.

quadrant. One of the four regions the x- and y-axes cut the plane into, numbered I, II, III, IV anticlockwise from the top-right.

quotient. the result of a division – how many times the divisor fits.

radian. The angle that cuts off an arc equal in length to the radius; a full turn is 2 pi radians.

radical. the square-root symbol; it returns the non-negative (principal) root

radius. the distance from the centre of a circle to its edge

radix. another word for base.

range. the set of outputs a function actually produces (the right-hand dots hit by an arrow)

Rate of change. How fast one quantity changes as another changes; the derivative read with real-world units.

ray. a part of a line with one endpoint that continues forever in one direction

reciprocal. The fraction turned upside-down; $\frac{a}{b}$ has reciprocal $\frac{b}{a}$, and a number times its reciprocal is 1.

recurrence. a rule that defines each term from earlier terms, together with one or more starting values

relation. a set of ordered pairs (a, b) recording which elements are linked

relatively prime. Two numbers whose only common factor is 1 (also called coprime).

remainder. what is left over after dividing – the part smaller than the modulus

Removable hole. A single missing point on an otherwise smooth graph, where the limit exists even though the function is undefined.

reverse percentage. Finding the original amount before a known percentage change, by dividing by the multiplier.

root. a value of x that makes the polynomial equal zero; on a graph, where the curve crosses the x-axis

rotation. turning a point about a fixed centre by an angle, keeping its distance from the centre unchanged

rounding. Replacing a number with a nearby simpler value to a chosen place, using the next digit to decide up or down.

row count 2^n. A statement with n distinct simple statements has 2^n rows, since each input is independently T or F.

scalar. an ordinary single number, used to stretch or shrink a vector

Second derivative. The derivative of the derivative, written $f''(x)$; its sign tells whether the curve is cupped up (a valley) or down (a hill).

sector. a "pizza slice" of a circle, bounded by two radii and an arc

semi-perimeter. half the perimeter, $s = (a + b + c)/2$, used in Heron's formula

sequence. an ordered list of numbers; a_n denotes the term in position n

series. the sum of the terms of a sequence

set. an unordered collection of distinct elements, written in braces $\{\}$

SI prefix. a metric multiplier that is a power of ten, such as kilo (10^3) or milli (10^{-3}).

sign. whether a number is positive or negative.

similar. the same shape but possibly different size (equal angles, sides in proportion)

solution. A value of the variable that makes the equation true; substituting it gives equal sides.

solution of a system. values for the unknowns that make every equation in the system true at once

solution set. the complete collection of values that make an inequality true

sound argument. a valid argument whose premises are in fact true

statement. A declarative sentence that is definitely either true or false (but not both); also called a proposition.

sum. the result of an addition (also called the total).

supplementary. two angles that add to 180°

surface area. the total area of all the faces (the outside skin) of a solid; in m²

system of equations. two or more equations sharing the same unknowns, to be satisfied together

tangent. a line that touches a circle at exactly one point

tautology. A statement whose truth-table column is all T – true under every assignment.

tenths. The first place to the right of the decimal point, worth $\frac{1}{10}$.

term. A number, variable, or product of them, separated from others by + or - signs.

terminating decimal. A decimal that ends after finitely many digits, such as 0.375.

transformation. a rule that maps every point of a shape to a new position

transistor. a microscopic electronic switch (off or on) that physically stores a bit.

translation. a slide: every point moves the same distance in the same direction

transversal. a line that cuts across two or more other lines

triangle. a closed shape with three straight sides and three angles summing to 180°

truth table. A grid listing the truth value of a compound statement for every combination (2^n) of its inputs.

truth value. The label – true (T) or false (F) – that a statement carries.

union. the set $A \cup B$ of elements in A **or** B (or both)

unit circle. The circle of radius 1 centred at the origin; a point at angle theta has coordinates (cos theta, sin theta).

unit fraction. a fraction equal to 1 whose numerator and denominator are the same quantity in different units

universal quantifier. $\forall$, "for all" – claims a predicate holds for *every* element of the domain

vacuously true. A conditional with a false hypothesis, which is automatically true because the promise was never tested.

valid argument. one whose form makes a false conclusion impossible when all premises are true

variable. A letter standing for an unknown or changing number.

vector. an arrow with a direction and a length, written (a, b): go a across and b up

Velocity. The rate of change of position with respect to time; speed together with a direction (a sign).

vertex. the common point where the two arms of an angle meet

vertically opposite. the equal angles formed opposite each other when two lines cross

volume. the amount of space a solid occupies; in cubic units, m^3

wave. the smooth, endlessly repeating up-and-down shape of the sine and cosine graphs

y-intercept. the value of y where a line crosses the y-axis (x = 0)

Appendix C – Symbol & Notation Reference

Symbol	Meaning
$+ \; - \; \times \; \div$	add, subtract, multiply, divide
$= \; \neq$	equal to, not equal to
$< \; \leq \; > \; \geq$	less / less-or-equal / greater / greater-or-equal
$a^n \; \sqrt{a}$	power, square root
$\wedge \; \vee \; \neg$	logical AND, OR, NOT
$\Rightarrow \; \Leftrightarrow$	implies, if-and-only-if
$\forall \; \exists$	for all, there exists
$\cup \; \cap \; \subseteq \; \in$	union, intersection, subset, element of
`== != and or not // % **`	Python operators for the above