COMPUTING AND DIGITAL LITERACY

# Basic Computational Problem-Solving

### Difficulty level: Intermediate

### Computing and Digital Literacy

This document was made by AI and there may be mistakes. If a question or answer looks wrong, use the Flag a mistake button to let Brett know so he can fix it.

## *Overview of the concept*

This is how you tackle a big messy task, like planning a school event or fixing a broken phone app. You break it into smaller parts, spot patterns, and write clear steps, which turns a problem that feels overwhelming into a series of small jobs. It is the thinking behind all programming, and it works just as well on everyday problems that have nothing to do with computers.



© 2026 Brett Robbins. All Rights Reserved.

## Worked examples

EXAMPLE 1

**Q** You have to plan a class party for 20 guests. How do you start?

SOLUTION

This is decomposition: break the big job into smaller parts.

Split it into guest list, food, games, invitations.

Then take the parts in order: list guests, choose food, plan games, send invites.

EXAMPLE 2

**Q** You need the fastest route to school. What do you keep, and what do you ignore?

SOLUTION

This is abstraction: hide the details that do not change the answer.

Keep what affects the answer: streets, distances, and travel time.

Hide the rest: shop names, house colours, pavement cracks.

Now the problem is small enough to solve: compare the routes on travel time alone.

© 2026 Brett Robbins. All Rights Reserved.

## Practice questions

**1** You must plan a school fair. Break it into four smaller parts.

- **A** One possible option: stalls, food, games, and tickets
- **B** Buy everything the day before the fair.
- **C** Stalls only
- **D** Ask the teacher to plan the whole fair.

Understand It with Pulse | Flag a mistake

© 2026 Brett Robbins. All Rights Reserved.

## 2 What is decomposition?

- **A** Hiding the details that do not change the answer
- **B** Writing clear steps for each part
- **C** Breaking a big problem into smaller parts
- **D** Reusing a known structure

Understand It with Pulse | Flag a mistake

© 2026 Brett Robbins. All Rights Reserved.

## 3 A problem repeats every week with the same structure. What helps?

- **A** Decomposition: break it into smaller parts.
- **B** Pattern recognition: reuse the known structure.
- **C** Abstraction: hide the details that do not matter.
- **D** Algorithms: write clear steps for each part.

Understand It with Pulse | Flag a mistake

© 2026 Brett Robbins. All Rights Reserved.

**4** **You must plan the cheapest way to get to the city. Which details matter, and which can you hide?**

A. Keep: ticket prices, travel times, and routes. Hide: the colour of the bus, the seat fabric, and the driver's name.

B. Keep: the colour of the bus, the seat fabric, and the driver's name. Hide: ticket prices, travel times, and routes.

C. Keep: ticket prices and the driver's name. Hide: travel times, routes, and the colour of the bus.

D. Keep: travel times and the seat fabric. Hide: ticket prices, routes, and the colour of the bus.

Understand It with Pulse | Flag a mistake

© 2026 Brett Robbins. All Rights Reserved.

## 5 What is abstraction?

- A Breaking a big problem into smaller parts
- B Hiding the details that do not change the answer
- C Reusing a known structure
- D Writing clear steps for each part

Understand It with Pulse | Flag a mistake

## 6 What is the last step after breaking a problem down?

- A Break the problem into smaller parts again.
- B Reuse the same structure.
- C Hide the details that do not matter.
- D Write clear steps (an algorithm) for each part and combine them.

Understand It with Pulse | Flag a mistake

© 2026 Brett Robbins. All Rights Reserved.

7

## Two problems look alike: planning a party and planning a school fair. What does that tell you?

A You can reuse the same structure.

B You must start from scratch each time.

C The two problems have nothing in common.

D You can only solve one problem at a time.

Understand It with Pulse

Flag a mistake

© 2026 Brett Robbins. All Rights Reserved.

8

## You must pick the best of three bicycles to buy. Which details matter, and which can you hide?

A. Keep: price and the shop's posters. Hide: size, brakes, and the colour of the box it came in.

B. Keep: size and the colour of the box it came in. Hide: price, brakes, and the shop's posters.

C. Keep: price, size, and brakes. Hide: the colour of the box it came in and the shop's posters.

D. Keep: the colour of the box it came in and the shop's posters. Hide: price, size, and brakes.

Understand It with Pulse

Flag a mistake

© 2026 Brett Robbins. All Rights Reserved.

9

## Name the four parts of computational thinking.

A. Decomposition, pattern recognition, abstraction, testing

B. Decomposition, pattern recognition, abstraction, algorithms

C. Decomposition, coding, abstraction, algorithms

D. Pattern recognition, abstraction, algorithms, debugging

Understand It with Pulse

Flag a mistake

© 2026 Brett Robbins. All Rights Reserved.

10

## You must work out the cost of a party. Which details matter, and which can you hide?

A. Keep: the colour of the invitations and the brand of the cups. Hide: number of guests, food cost, and room cost.

B. Keep: number of guests and the brand of the cups. Hide: food cost, room cost, and the colour of the invitations.

C. Keep: food cost and the colour of the invitations. Hide: number of guests, room cost, and the brand of the cups.

D. Keep: number of guests, food cost, and room cost. Hide: the colour of the invitations and the brand of the cups.

Understand It with Pulse | Flag a mistake

© 2026 Brett Robbins. All Rights Reserved.

## ANSWER KEY

1. One possible option: stalls, food, games, and tickets

2. Breaking a big problem into smaller parts

3. Pattern recognition: reuse the known structure.

4. Keep: ticket prices, travel times, and routes. Hide: the colour of the bus, the seat fabric, and the driver's name.

5. Hiding the details that do not change the answer

6. Write clear steps (an algorithm) for each part and combine them.

7. You can reuse the same structure.

8. Keep: price, size, and brakes. Hide: the colour of the box it came in and the shop's posters.

9. Decomposition, pattern recognition, abstraction, algorithms

10. Keep: number of guests, food cost, and room cost. Hide: the colour of the invitations and the brand of the cups.

© 2026 Brett Robbins. All Rights Reserved.