

Algorithmic Thinking: Decompose & Test

Break a complex task into smaller steps, specify an algorithm and test edge cases.

LEARN • PRACTISE • CREATE • REFLECT

Original BrightPencil Kids resource - adult-led where appropriate

A4 edition • 12 purposeful pages • review copy

Print. Learn. Grow.

Grown-up guide

Learning objective

Break a complex task into smaller steps, specify an algorithm and test edge cases.

Use this pack well

- Model the first example aloud, then let the learner try.
- Ask “How do you know?” more often than “Are you finished?”.
- Use real objects, talk, movement or observation when the activity suggests it.
- Stop or shorten the session if attention or enjoyment drops.
- Open-ended pages can have many successful answers.

Quality note

This resource was written originally for BrightPencil Kids. Curriculum sources informed scope and progression only; no third-party worksheet or story text is reproduced.

Learn the idea

1 Decomposition reduces a large problem into manageable parts.

2 Algorithms need unambiguous steps and conditions.

3 Edge cases test situations near limits or unusual inputs.

Talk about it

Explain one idea in your own words. Give an example from real life, a drawing or an object if that helps.

Build a clear model or organiser that shows: Decomposition reduces a large problem into manageable parts.

Use supplied evidence to test this idea: Algorithms need unambiguous steps and conditions.

Compare two cases and explain where this principle matters: Edge cases test situations near limits or unusual inputs.

Find one weak claim or assumption and revise it so it matches the evidence.

Write a short conclusion that states evidence, limitation and next question.

Try it beyond the page

Real-world / hands-on connection

Turn a school-safe task into ordered steps and conditions.

Before you finish

What did you notice?

What evidence, object or example helped you?

What would you change if you tried again?

Create something

Your project

Create an algorithm for sorting fictional library cards and test three edge cases.

Use this space for planning, drawing, mapping, designing or notes.

CHALLENGE

Think deeper

1. What is decomposition?

2. Why test edge cases?

3. What makes an algorithm executable?

Quick review

1. In one sentence, explain the big idea of “Algorithmic Thinking: Decompose & Test”.

2. Draw or write one example that shows you understand.

3. Choose one activity from this pack and explain the strategy you used.

4. Write one question you still have.

Answer & discussion notes

Use these as guidance, not scripts. Open questions can have different successful responses when the reasoning is sensible and evidence-based.

1. Suggested response

Breaking a problem into smaller parts.

2. Suggested response

They can reveal failures hidden by ordinary examples.

3. Suggested response

Clear, ordered and unambiguous instructions.

Review before publication

Check age fit, factual accuracy, clarity, accessibility, rights provenance and whether the activity works in practice. This file remains a review copy until those checks are signed off.

FINISH

My learning reflection

I finished this BrightPencil learning pack.

One thing I understand better now:

A strategy that helped me:

Something I created or noticed:

My next small question or goal:

Print. Learn. Grow.