

DTiF

Digital Technologies in focus

Initiative of and funded by the Australian Government Department of Education, Skills and Employment

acara AUSTRALIAN CURRICULUM,
ASSESSMENT AND
REPORTING AUTHORITY



COMPUTATIONAL THINKING IN PRACTICE

Parent and teacher cards Years F–2

Computational thinking is a core concept in the Australian Curriculum: Technologies. It includes:

- organising information (data) logically
- breaking down problems into parts
- defining abstract concepts
- designing and using algorithms (step-by-step instructions)
- understanding patterns and creating models.

For more information see: <https://www.australiancurriculum.edu.au/f-10-curriculum/technologies/key-ideas/>

Many everyday tasks that require a step-by-step approach or learning a routine involve computational thinking. Examples include following a recipe, getting dressed and ready for school in the morning, and following the rules of a game.

Parents and teachers can help children in the early years (F–2) to build understanding and find opportunities to apply computational thinking when carrying out simple everyday tasks.

Computational thinking is an invaluable approach to problem-solving in all learning areas as students complete primary education and especially as they move into the secondary years.

Teachers could use the following cards as a resource for parents to do activities at home



All images in this resource used with permission

COMPUTATIONAL THINKING

Look for patterns at home.

Patterns help us solve problems.

Repeating patterns help to make tasks easier.

What is the pattern you use to set the table to eat?



PATTERN RECOGNITION

Analyse the data and look for patterns that make sense of the data or problem

COMPUTATIONAL THINKING

Look for patterns at home.

Patterns help us solve problems.

Repeating patterns help to make tasks easier.

Where can you see patterns at home?



PATTERN RECOGNITION

Analyse the data and look for patterns that make sense of the data or problem



COMPUTATIONAL THINKING

Think about how games have steps (algorithms) that need to be followed.

Games use patterns to repeat play and to keep score.

**What algorithm can you think of for a favourite game?
How would you keep score?**



ALGORITHMS

Create a series of ordered steps that solve a class of problem



PATTERN RECOGNITION

Analyse the data and look for patterns that make sense of the data or problem

COMPUTATIONAL THINKING

Look for patterns at home.

Patterns help us remember things we need to repeat.

Loops are patterns that repeat. We find them in music.

Describe the loops in the song *Happy Birthday*.



PATTERN RECOGNITION

Analyse the data and look for patterns that make sense of the data or problem



COMPUTATIONAL THINKING

Describe the algorithm for how you get to school step by step.

An algorithm is a plan or set of step-by-step instructions used to solve a problem.

Draw a map showing the way to a special place.



ALGORITHMS

Create a series of ordered steps that solve a class of problem

COMPUTATIONAL THINKING

Think of a new game and tell a friend the rules of play.

An algorithm is a plan or set of step-by-step instructions used to solve a problem.

Explain the steps to play a game of 'Simon says' with your friends or family.



ALGORITHMS

Create a series of ordered steps that solve a class of problem



COMPUTATIONAL THINKING

Describe the steps to make a recipe or favourite meal.

An algorithm is a plan or set of step-by-step instructions used to solve a problem.

Help an adult to follow instructions to cook a meal.



ALGORITHMS

Create a series of ordered steps that solve a class of problem

COMPUTATIONAL THINKING

Describe your morning routine step by step.

An algorithm is a plan or set of step-by-step instructions used to solve a problem.

Explain what you do first when you wake up. What do you do next? How many steps are there to start your day?



ALGORITHMS

Create a series of ordered steps that solve a class of problem



COMPUTATIONAL THINKING

Explain to an adult how to build a house with building blocks or a cubby house.

An algorithm is a plan or set of step-by-step instructions used to solve a problem.

Give an adult the instructions to build a cubby or a den, inside the house or outdoors.



ALGORITHMS

Create a series of ordered steps that solve a class of problem

COMPUTATIONAL THINKING

Look for patterns where you live.

Patterns help us solve problems.

Repeating patterns helps to make tasks easier.

Collect some sticks and leaves from outside – can you make a repeating pattern with them?



PATTERN RECOGNITION

Analyse the data and look for patterns that make sense of the data or problem



COMPUTATIONAL THINKING

Teach a family member how to make a paper plane.
Test them ... Whose plane flew the furthest?

An algorithm is a plan or set of step-by-step instructions used to solve a problem.

Describe all the steps to make a paper plane.



ALGORITHMS

Create a series of ordered steps that solve a class of problem

COMPUTATIONAL THINKING

Look for algorithms in the rules of games.

An algorithm is a plan or set of step-by-step instructions used to solve a problem.

Play a card or board game with your family. Make sure everyone follows the rules!



ALGORITHMS

Create a series of ordered steps that solve a class of problem



COMPUTATIONAL THINKING

Learning how to play a game is easier in small steps.

Decomposition involves breaking down a big problem into smaller parts that are easier to manage or understand.

Describe the rules for a sport or activity that you do. How do you start? What is the next step?



DECOMPOSITION
Break problems into parts

COMPUTATIONAL THINKING

Learning how to brush our teeth is easier in small steps.

Decomposition involves breaking down a big problem into smaller parts that are easier to manage or understand.

Explain step by step how to brush your teeth or wash your hands. What do you do first? What do you do next?



DECOMPOSITION
Break problems into parts



COMPUTATIONAL THINKING

Abstraction is the process of making something easier to understand by concentrating only on the most important idea. For example, we can describe a story by saying it's about helping your friends, without saying everything that happens.

Describe a favourite movie in a few sentences. Is it a happy or sad story? What is the main idea of the movie? Friendship? Adventure?



ABSTRACTION

Remove unnecessary detail and focus on important aspects of structure or data

COMPUTATIONAL THINKING

Abstraction is the process of making something easier to understand by concentrating only on the most important idea. For example, we can draw a line curved up at both ends to show happiness, without drawing the details of either a face or a person.

What shapes could you use to draw a house or tree?



ABSTRACTION

Remove unnecessary detail and focus on important aspects of structure or data



COMPUTATIONAL THINKING

Modelling and simulation are ways we can show another person information or processes. For example, we can show how long something is with footsteps.

Make a model that shows how tall you are with blocks.



MODELS AND SIMULATIONS

Create and apply models or simulations that represent situations

COMPUTATIONAL THINKING

Modelling and simulation are ways we can show another person information or processes. For example, we can learn the steps to wash our hands by watching someone do this or by seeing it acted out.

Describe the steps you take to wash your hands as you are doing it.



ALGORITHMS

Create a series of ordered steps that solve a class of problem



MODELS AND SIMULATIONS

Create and apply models or simulations that represent situations



Computational thinking



Computational thinking helps people to organise data logically by breaking down problems into parts; defining abstract concepts; and designing and using algorithms, patterns and models.

