

Introduction to Fundamental Concepts

IN 1101 PROGRAMMING FUNDAMENTALS

Computational Thinking

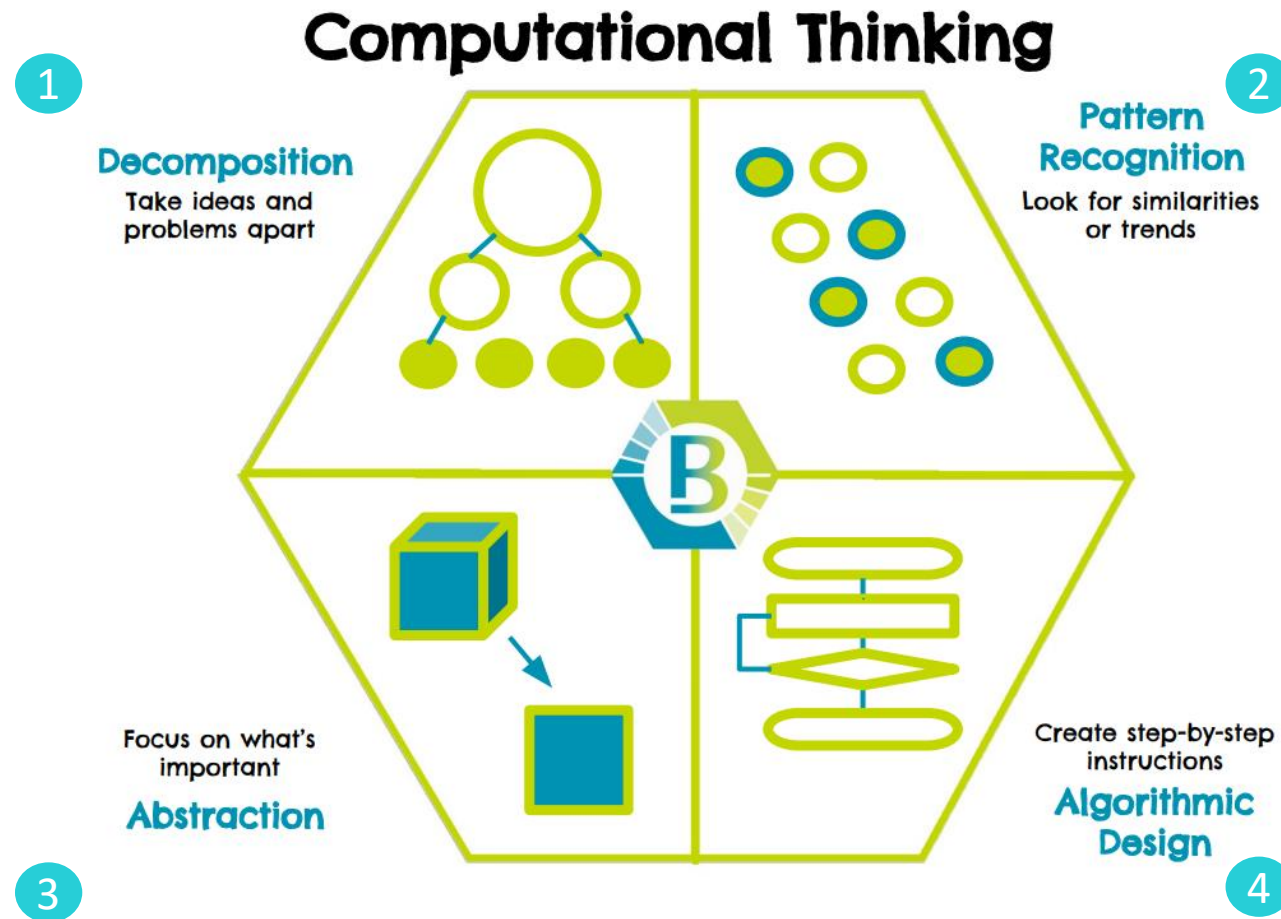


Source: <https://bsd.education/l/computational-thinking/>

Problem Solving

- ❑ **Computational Thinking**: A way of thinking for logically and methodically solving problems using concepts and ideas from computer science.
- ❑ Computational thinking applied in,
 - Engineering
 - Medicine
 - Agriculture
 - Finance
 - And everywhere...

Key Techniques



Source: <https://www.wcpss.net/domain/17003>

Key Techniques Cont..

☐ **Decomposition.**

- Breaking down a complex problem or system into smaller, more manageable parts.

☐ **Pattern Recognition.**

- Looking for similarities among and within problems.

☐ **Abstraction.**

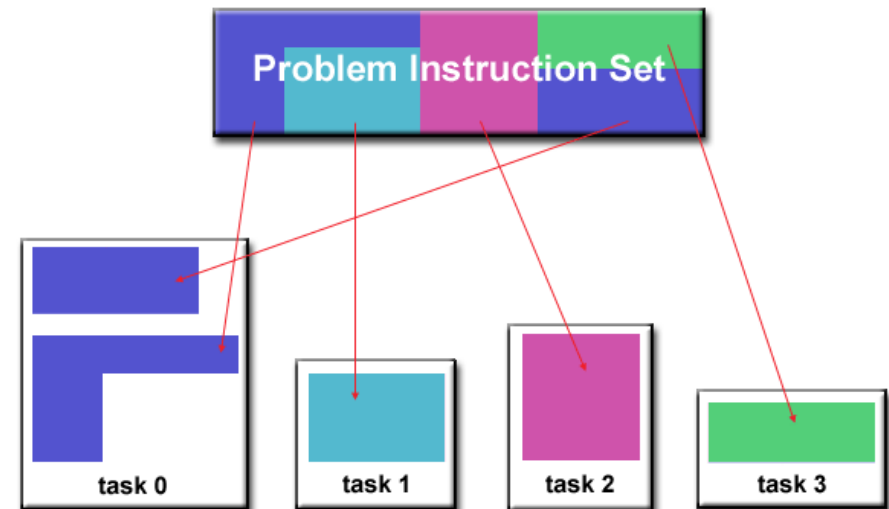
- Focusing on the important information only.

☐ **Algorithm Design.**

- Developing a step-by-step solution to the problem.

Key Techniques: Decomposition

- ❑ Breaking down a complex problem or system into smaller, more manageable parts.
- ❑ It will allow to understand and execute the process much easily.
- ❑ How to decompose?
 - ❑ Divide a task into a sequence of subtasks.
 - ❑ Identify elements or parts of a complex system.



Key Techniques: Pattern Recognition

- Noticing or identifying similarities or common differences that will help us make predictions or lead us to shortcuts.
- Based on experience, we develop shortcuts mapping problem characteristics to solution.
 - We look for *patterns* when we play games to decide when to do certain things.



Figure 1



Figure 2

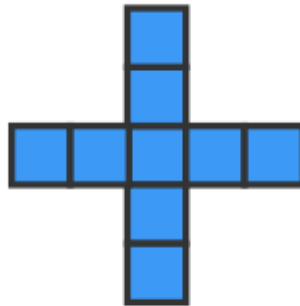


Figure 3

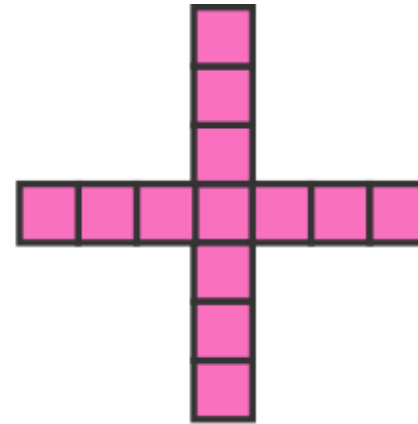
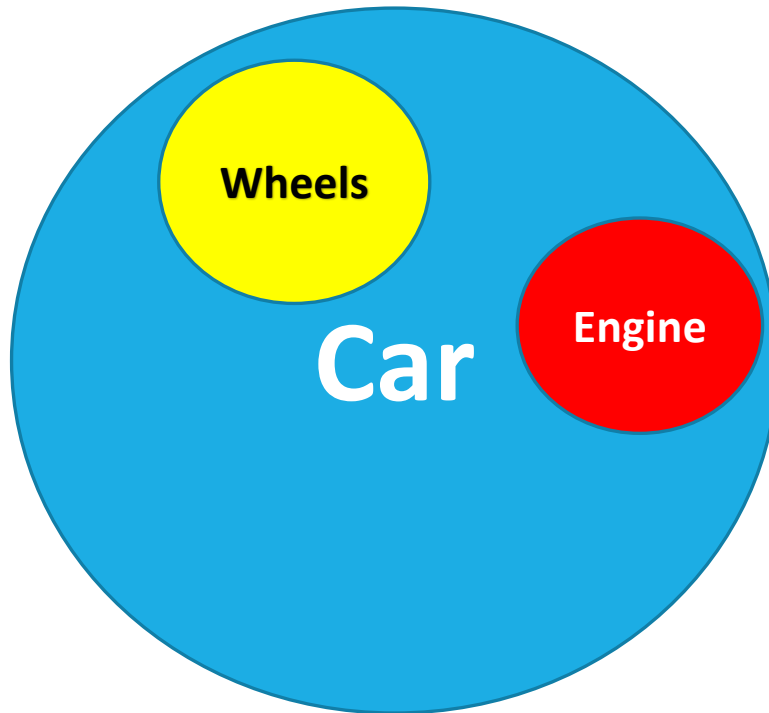


Figure 4

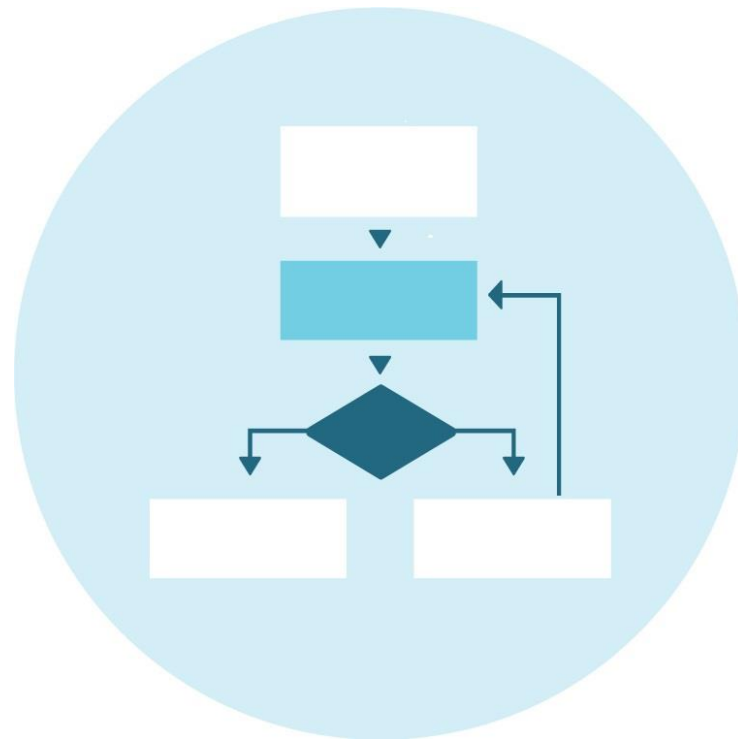
Key Techniques: Abstraction

- ❑ Focusing on the important information only, ignoring irrelevant detail.
- ❑ A “Big Picture”.



Key Techniques: Algorithm Design

- ❑ Developing a step-by-step strategy for solving a problem.
- ❑ Algorithmic thinking involves both creation and execution of an algorithm.



Understanding Algorithms

- ❑ An algorithm is a set of step-by-step instructions to solve a problem.
- ❑ We often use algorithm in our day-today life.

E.g. :

- Cooking recipe.
- Buy something from shop.
- Withdraw money from ATM.
- Make a cup of tea.
- Daily routing as a student.

Activity 1

Use one of following problems and write how you could solve using computational thinking.

(hint: Use key techniques)

1. Late to go to work in the morning.
2. Planning trip to Nuwara Eliya.
3. Planning school sports meet.

How to Write Algorithms?

- ❑ Define **inputs** of the algorithm
 - ❑ E.g. To calculate the area of rectangle input may be the rectangle height and rectangle width.
- ❑ Define the **variables**
 - ❑ E.g. We can define two variables for rectangle height and rectangle width as HEIGHT and WIDTH.
- ❑ Outline the **operations**
 - ❑ E.g. To find area of rectangle multiply the HEIGHT and WIDTH variable and store the value in new variable.
- ❑ **Output** the results of the operations
 - ❑ E.g. In case of area of rectangle output will be the value stored in variable AREA

How to Represent Algorithms?

- ❑ **Flowchart** – A graphical representation using standardized symbols.
- ❑ **Pseudocode** – Textual representation in English –like phrases.

```
Start  
  Read length  
  Read width  
  area = length x width  
  Display area of a rectangle  
End
```

Figure: Sample Pseudocode

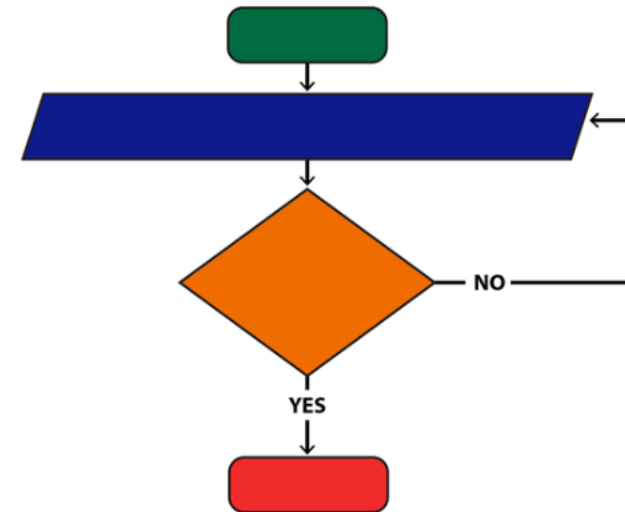




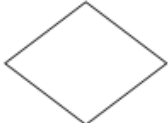

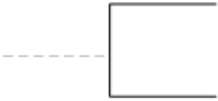


Figure: Sample Flowchart

Flowcharts

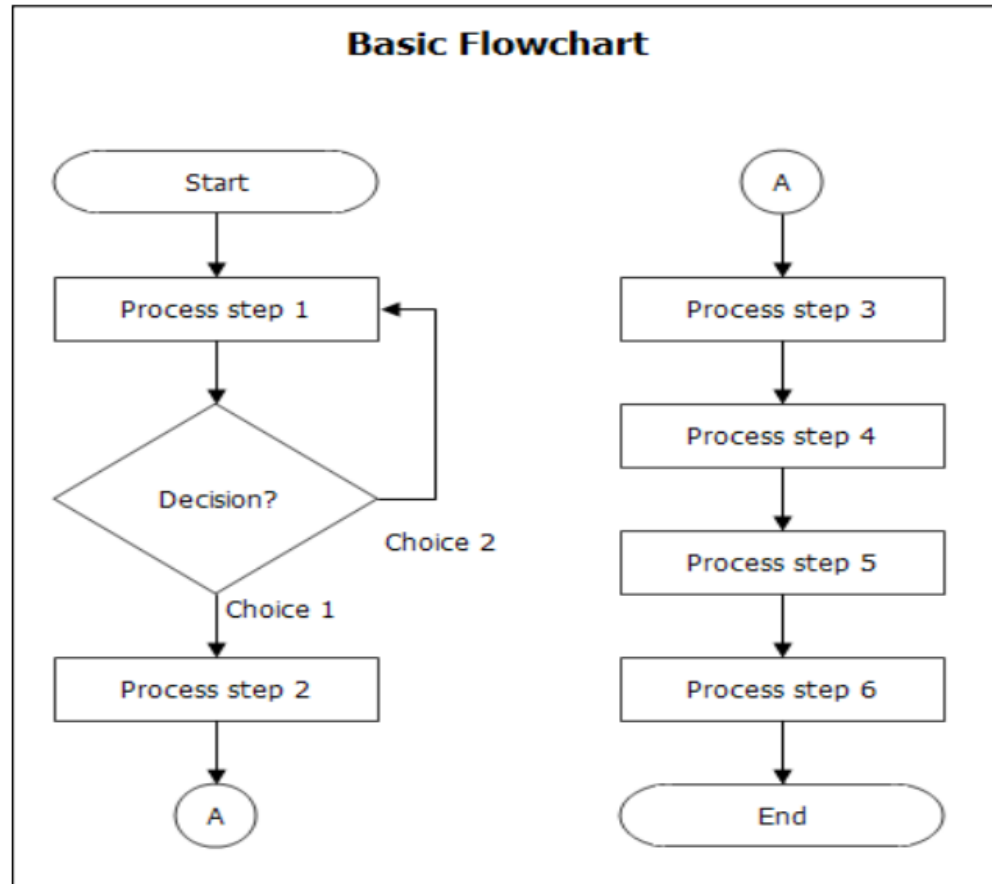
- ❑ Uses different symbols to design a solution to a problem.
- ❑ By looking at a flowchart, one can understand the operations and their sequence in solving a problem.
- ❑ Advantages:
 - ❑ Easy and Efficient to analyze problem.
 - ❑ It is easy to convert the flowchart into any programming language.

Flowchart Symbols

Flowchart Symbol	Symbol Name	Description
	Terminal (Start or Stop)	Terminals (Oval shapes) are used to represent start and stop of the flowchart.
	Flow Lines or Arrow	Flow lines are used to connect symbols used in flowchart and indicate direction of flow.
	Input / Output	Parallelograms are used to read input data and output or display information
	Process	Rectangles are generally used to represent process. For example, Arithmetic operations, Data movement etc.
	Decision	Diamond shapes are generally used to check any condition or take decision for which there are two answers, they are, yes (true) or no (false).
	Connector	It is used connect or join flow lines.
	Annotation	It is used to provide additional information about another flowchart symbol in the form of comments or remarks.

Note: There are many more notations.

Flowcharts Cont..



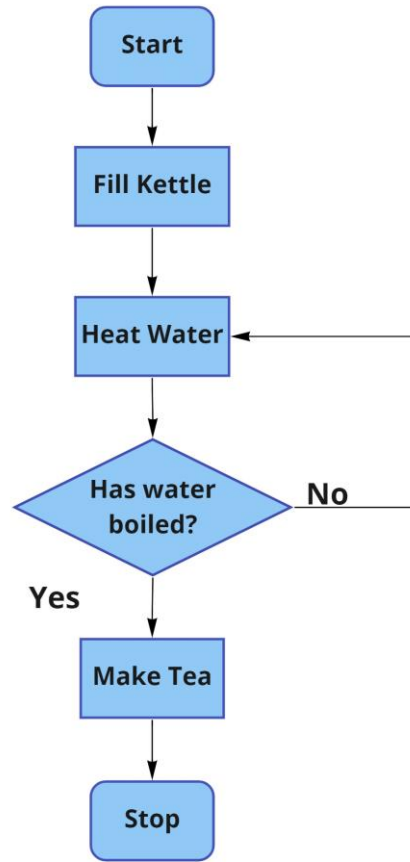
Flowcharts: Example 1

Problem: Make a cup of tea



Flowcharts: Example 1

Making Tea

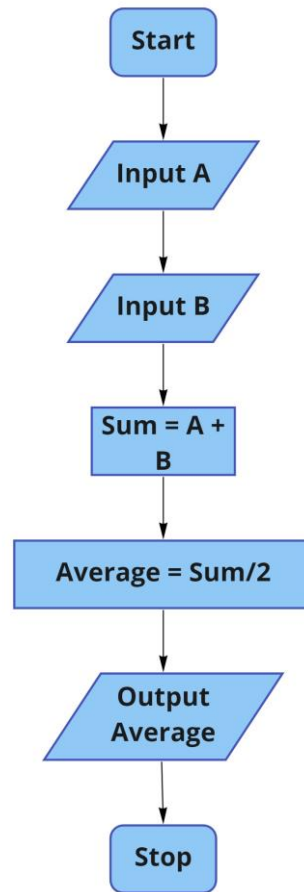


Note: You can have multiple flowcharts for the same problem based on how you decompose the problem.

Flowcharts: Example 2

- Draw a flow chart to solve the following problem.
 - Input : Two numbers
 - Output: Average of two input numbers.

Flowcharts: Example 2



Activity 2

Draw a flow chart to solve the following problem.

1. Display largest number from two inputs.
2. Display the area of a circle from input radius R .

Home work

1. Display the sum of numbers between 1 to 10.
2. Print multiplication table of a number.

Pseudo Codes

- ❑ A simple way of writing programming code in English.
- ❑ Algorithms will often be expressed in pseudo code.
- ❑ Describes the logic of algorithm.
- ❑ Not actual programming language.

Pseudo Codes: Common Action Words

➤ Input:

- READ
- OBTAIN
- GET

➤ Output:

- PRINT
- DISPLAY
- SHOW

➤ Compute/Process:

- COMPUTE
- CALCULATE
- DETERMINE

➤ Initialize:

- SET
- INIT

➤ Add one:

- INCREMENT
- BUMP

Pseudo Codes: Example 1

Let's write the pseudo code for making a cup of tea.

```
BEGIN
  Fill kettle with water
  Heat water
  IF water boiled
    Make Tea
  Else
    Heat water
END
```


Pseudo Codes: Example 2

Write the pseudo code to print the average of two input numbers.

BEGIN

READ A

READ B

CALCULATE the summation of A and B

CALCULATE the summation/2 as average

DISPLAY average

END

Activity 3

Write pseudo codes for the problems mentioned in Activity 2.

Questions?