

Hi, welcome to Basic Program Structure.

## ☀ **Flowchart and Pseudo-code**

You already learnt that Algorithm is used to describe the solution to a problem. How do you express an algorithm?

☀ Program logic may be represented by flowcharts or pseudocode. Both can be used in the design of a program. A flowchart is a way to represent the program logic or algorithm by a diagram. It describes the logic of the operations and the sequence in which they are carried out by the program. Pseudocode uses English-like statements to design and describe the operations performed by the program. Both flowcharts and pseudocode are useful tools to help us determine the order of the operations to solve a problem. They are also good tools for documentation. At the end of this lesson, you should be able to:

Apply top-down design methodology to develop an algorithm to solve a problem

Express the solution(s) of a problem in such a way that a computer—human or machine—can effectively carry out

Express an algorithm using

Flowcharts and Pseudo-code

☀ It is easy to develop programs directly from flowcharts or pseudocode. In this lesson, we will describe both flowcharts and Pseudocode in details.

☀ Consider a common example in real life, (mouse move to middle of the screen and click) between two coffee shops, find the nearer one to meet a friend. How do you design an algorithm to solve the problem?

☀ The technique that is used for developing a program is the top-down stepwise technique. It uses the divide and conquer strategy. It consists of two techniques: decomposition and stepwise refinement.

Decomposition splits a large problem into a series of smaller subproblems. Then consider each subproblem separately and further split them into subproblems. To design an algorithm for this problem, we first decompose the algorithm comprising the following sequential steps:

Calculate the distances to locations

Find the nearer location

A structured program can be written using simple control structure to solve a problem. The simplest control structure is sequence structure.

A sequence structure contains a sequence of statements (or operations) that are performed one after another.

☀ When we formulate a method/ procedure for solving a problem, it has to be computable . So further refinement is necessary for each subtask of the program. We refine the steps of the problem solutions to each subproblem in greater detail until no further refinement is possible. The refinement process stops when all steps are detailed enough so that they can be directly translated into a program in a specific programming language.

☀ Both flowcharts and pseudocode can be used in the design of a program. We start from flowchart.

In flowcharts, the flow of the control can be easily visualized. A flowchart is composed of a set of standard symbols. The symbols are connected by flowlines.

The table shows some of the common symbols used in constructing flowcharts.

The terminal symbol, which is oval-shaped, is used to indicate the beginning or end of a program. It can also be used to specify the beginning or end of a function. The terminal symbol containing the word “start” is the first symbol used in a flowchart in a program; and the terminal symbol containing the word “End” is the last symbol used.

The flowlines are used to connect symbols and indicate the direction of processing.

The input/output symbol, which is represented by parallelogram symbol, is used to indicate the input/output operations.

The process symbol, which is a rectangle symbol, is used to indicate any type of processing or calculation. Process symbols are normally used to represent a collection of statements that perform the calculation.

The decision symbol, which is a diamond symbol, is used to indicate that a decision is to be made at a certain point in a flowchart at which a branch to one or more alternative paths is possible. It contains a condition that can either be true or false. Two flowlines leaving the corners of the diamond are labeled based on the condition in the symbol. One indicates the direction to be taken when the condition is true, and the other indicates the direction to be taken when the condition is false.

☀ A sequence structure contains a sequence of statements (or operations) that are performed one after another.

Flow chart may include Branching if making selection is necessary.

A selection structure contains a set of statements that is performed if a condition is true, and another set of statements that is performed if the condition is false. The simple if-else structure is shown here.

We add if-else structure into our flowchart.

☀️ Let's make the problem more complex. **What if there were many coffee shops? How to find the nearest one among them?**

☀️ If using a sequence structure, the flow chart is too long to manage, so the flowchart may include Looping that repeats certain operations.

A repetition structure contains a set of statements that is repeated as long as a condition is true. In this loop structure, a test condition is used to control the number of times the loop is going to repeat. The loop will be executed until the condition is not satisfied.

☀️ After replacing sequence structure by looping structure, the flowchart is much more manageable. As for the detailed implementation, you will learn later.

☀️ Another method to describe the program logic is pseudocode. The idea is to directly use informal English to describe an algorithm step by step with one step per line. It uses the structural conventions of a normal programming language, but is intended for human reading rather than machine reading.

☀️ There are no strict rules in writing pseudocode for program logic. Pseudocode is an informal language. It uses a mix of English, together with a set of special keywords to describe the operations of a program. Examples of some of the commonly used keywords are IF, ELSE, WHILE, READ, PRINT, INITIALIZE, COMPUTE, ADD, SUBTRACT.

Pseudocode usually starts an operation sentence with a verb such as READ to read an input PRINT to print an output.

And **INITIALIZE, COMPUTE, ADD, SUBTRACT, which** can also be used depending on the nature of the statements.

☀️ **A quick recap of Guidelines of Pseudocode.**

- *Write one statement per line only*
- *Capitalize the keywords*
- *Indent to show hierarchy*
- *End multi-line structures*
- *Keep statements programming-language independent*

☀️ The flowchart and Pseudocode for the logic design of the algorithm of scenario 1 are shown in this slide. You might find that these Pseudocode solutions are like recipes.

☀️ Just as a recipe is a set of instructions for cooking a particular food dish.

☀️ A pseudocode algorithm is a set of instructions for how to solve a particular problem.

☀️ Let's represent the same recipe using a flowchart, which looks more like computational thinking. Both flowcharts and pseudocode are useful tools. A flowchart is a way to visualize a diagram. Pseudocode uses English-like statements. You are required to master both.

☀️ Quick Check: Is an algorithm readable by computers?

☀️ No. Algorithm is used to describe the solution to a problem, which is programming language-independent and cannot be understood by computers.

☀️ Quick summary: Program design is one of the most important steps in the program development process. It aims to design the logic of the program or algorithm. When we write an algorithm for solving a given problem, we need to design our algorithm to follow the following rules.

an algorithm must be unambiguous. Every step of the algorithm must be clear and precise an algorithm must also specify the order of steps precisely We need to consider all possible points of decision, and for each decision point, we need to consider a step for each possible outcome. Finally, the algorithm must execute the steps and terminate in finite time.

In subsequent lessons, we will cover how to translate a flowchart or pseudocode solution into a program that can be run on a computer.

I hope you enjoy this lecture.

And, I will see you next time.