

# **SPM 2102**

# **PROGRAMMING LANGUAGE 1**

## **PROGRAMMING CYCLE**

By

**NORAH MD NOOR**



# Review

- + What computer programming?
    - = Set of instructions to solve the problem by computer system
  
  - + What the problem?
    - = Such as to calculate income tax, sales record, daily payment
-

# Review

To be consider by programming :

- What the final product?
- What the media for output / final product?
- How to make the decision in programming? How about programming input?
- That input can be used by computer?

*All these question will be discover through this subject discussion*

# Programming Cycle

Basically, there are **5 phases** involved :

1. Analysis & Problem Definition phase
  2. Logic & Algorithm Construction phase
  3. Language or Coding Production phase
  4. Testing & Implementing phase
  5. Maintenance & Documentation phase
-

# Programming Cycle

## Analysis

- Define the problem
- Output specification
- Input

## Logic

- Design the algorithm
- Programming logic

## Coding

- Coding the programming

## Testing / Debugging

- Programming testing
- Run the program

## Maintenance

- Modified for the program
- “Programming” documentation



# Programming Cycle : Analysis

1. Analysis and Problem Definition phase
  - What do you know about ANALYSIS?
  - What about PROBLEM DEFINITION?

# Programming Cycle : Analysis

- The problem will be stated clearly before any solution is obtained.
  - Eg : Could the problem be solved using a computer?
- Two things will be identified:
  - Input, data that is required to be put in to a computer.
  - Output, the outcome/result from the input given.

# Programming Cycle : Analysis

- The format on how the results will be displayed is also determined in this phase.

(eg : text, chart, table, pie chart, etc..)



# Design phase

- The problem solving designing process is also known as **algorithm construction (logic)**.
- Usually, algorithm will be constructed using the top-down approach, where the problem will be splitted into smaller sub-problems making it easier to be executed.

# Design phase

- Identified grade (A,B,C,D,E) for every mark (99 – 0)
- Sort the data (Input) by mark
  - 99 – 80 = A
  - .....
- Output? Let say 95, then A

# Design phase

- Designing algorithm for a programming problem will involve the usage of design tools, they are:
  - Pseudo Code
  - Flow Chart

# Production phase

1. Programmers will perform coding in this phase.  
It has 2 steps :
    - Interpreting algorithm into programming codes
    - The program will be compiled and executed
  2. In this phase, thorough selection for the most suitable programming language will be done.
-

# Testing and Implementing phase

1. Tests will be done to ensure that the program developed :
  - will solve the problem as planned by the programmer.
  - has no error / bugs / logic error / syntax error
  - debug

# Testing and Implementing phase

2. Basically, there are two common errors :
    - Syntax error (coding). Eg : mistaken instruction; printf, scanf, comma etc.
    - Logical error : referring to programming logic. Eg : add / subtract symbol (+/-) etc.
-

# Maintenance and Documentation phase

1. Documenting refers to collecting all information regarding/related to the program by the programmer, from the start to the testing phase.
  
2. Two types of documentation :
  - Internal documentation – direct comments in the programming codes
  - External documentation – anything related to the program; title, sketches, flow chart, output format etc.

# Maintenance and Documentation phase

3. Error rectifying must be done if there is still logical error that is untraceable during the testing phase.
4. Modification / addition occurs if users demand for changes/upgrades from time to time.



# ALGORYTHM DESIGNING TOOLS

# DESIGN TOOLS : Flow Chart

- A flow chart is a logical illustration of a program, a graphical description that is connected using arrows.
  - It describes the phases involved in the problem solving before developing the actual program.
  - It is built up of graphical symbols that each of them represents a specific instruction within the program.
-

# DESIGN TOOLS : Flow Chart



- The symbols in a flow chart must be arranged and coordinated to form an algorithm.
- Flow chart template can be used as a guidance to help programmers build a flow chart

# DESIGN TOOLS : Flow Chart

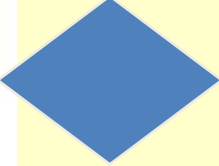

There is no best way to develop a flow chart, but there are some questions that programmers could answer in order to help develop their flow chart, as follows :

- What is the output that we want?
- What is the input that is needed?
- What is the process is involved?
- Is the problem about making a choice/decision making?
- Is the process involved have to be executed more than once?

# DESIGN TOOLS : Flow Chart

	Terminal	Starting (“start”) or ending (“end”) the algorithms.
	Data Input/Output	Input & Output

# DESIGN TOOLS : Flow Chart

	Testing	Usually used (If) term. Proceed the flow if (True) correct or accepted and go to others choice if (False)
	Process	Show the Operation such as Math operation and define pre-value

# DESIGN TOOLS : Flow Chart



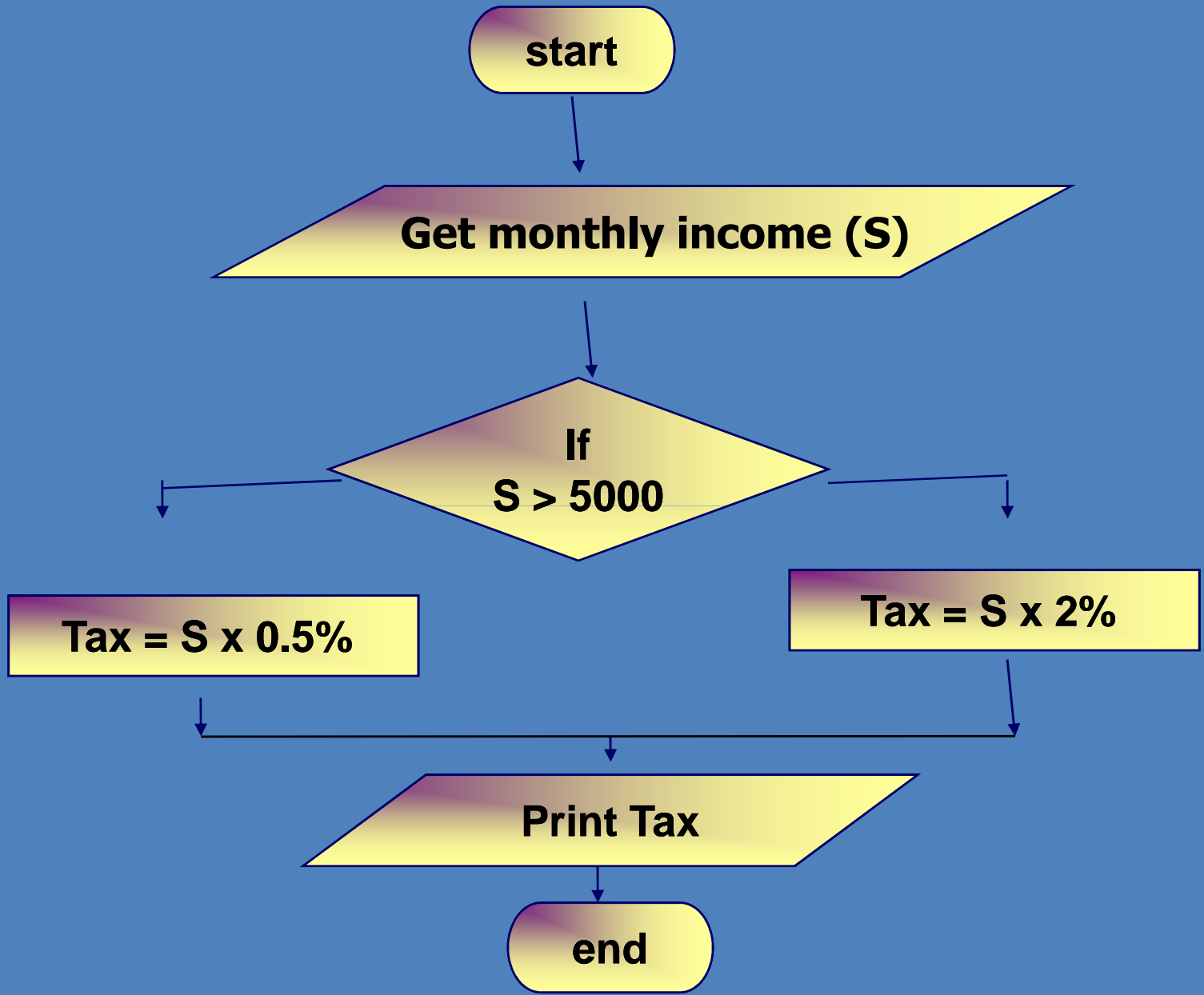
Connector

Connecting flows within same or different page



Arrow

To connect between symbols

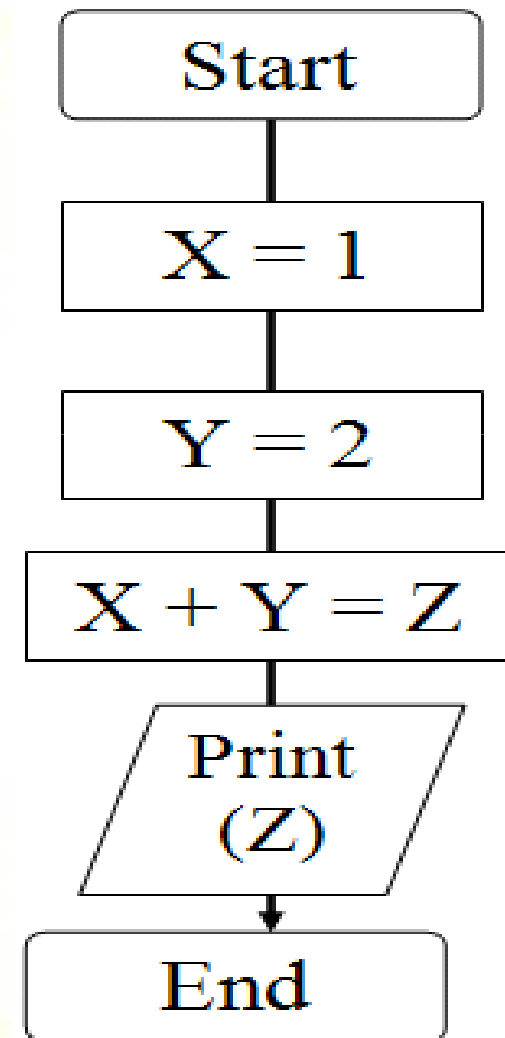




# Structured Programming

## Pseudo Code

1. Input  $X = 1$
2. Input  $Y = 2$
3. Calculate  $X + Y = Z$
4. Output / Print  $Z$
5. End



END