CS118 Programming Fundamentals

LAB 02

PRE-PROGRAMMING PHASE

NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES

What Problem Can Be Solved By Computer

- When the solution can be produced by a set of step-by-step procedures or actions.
- This step-by-step action is called an algorithm.
- The algorithm will process some inputs and produced output.
- Solving problem by computer undergo two phases:
  - **Phase 1:**
    - Organizing the problem or pre-programming phase.
  - **Phase 2:**
    - Programming phase.

PRE-PROGRAMMING PHASE

This phase requires five steps:

1. Analyzing the problem.
2. Developing the Hierarchy Input Process Output (HIPO) chart or Interactivity Chart (IC).
3. Developing the Input-Process-Output (IPO) Chart.
4. Drawing the Program flowcharts.
5. Writing the algorithms

I. Analyzing The Problem

- Understand and analyze the problem to determine whether it can be solved by a computer.
- Analyze the requirements of the problem.
- Identify the following:
  - ✓ Data requirement.
  - ✓ Processing requirement or procedures that will be needed to solve the problem.
  - ✓ The output.
- All These requirements can be presented in a Problem Analysis Chart (PAC)

Data	Processing	Output
Given in the problem or provided by the user.	List of processing required or procedures.	Output requirement.

Example # 01:

Write a Problem Analysis Chart (PAC) to find an area of a circle where

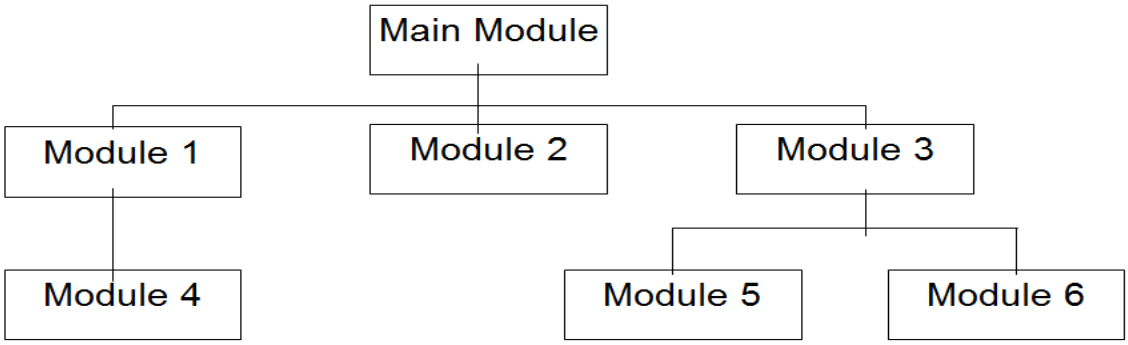
area = pi * radius * radius

Data	Processing	Output
radius	area = 3.14 x radius x radius	area

II. Developing the Hierarchy Input Process Output (HIPO) or Interactivity Chart

- The problem is normally big and complex.
- Thus, requires big program.
- Thus, the processing can be divided into subtasks called modules.
- Each module accomplishes one function.
- These modules are connected to each other to show the interaction of processing between the modules.
- Main/control module controls the flow all other modules.
- The IC is developed using top-down-method: top to down left to right order (also refer to order of processing).
- Modules are numbered, marked for duplication, repetition or decision.

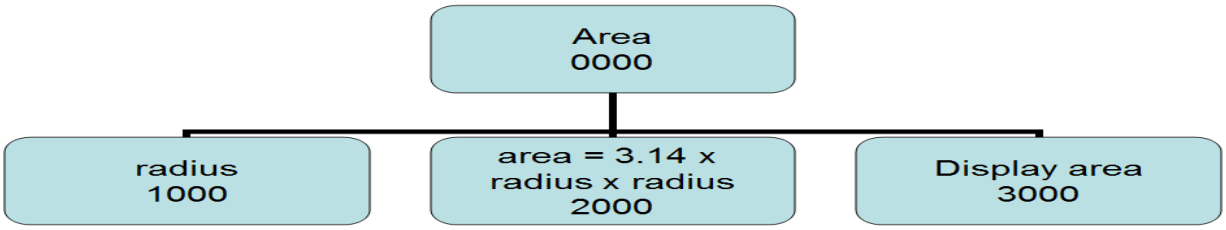
The interaction will form a hierarchy, called Hierarchy Input Process Output Chart (HIPO) or Interactivity Chart (IC). Programming which use this approach (problem is divided into subtasks) is called ***Structured Programming***.



Example # 02:

Write a Hierarchy Input Process Output (HIPO) to find an area of a circle where

area = pi * radius * radius



III. Developing the Input Process Output (IPO) Chart

- Extends and organizes the information in the Problem Analysis Chart.
- It shows in more detail what data items are inputs, what is the processing or modules on that data, and what will be the result or output.
- It combines information from PAC and HIPO Chart.

Example # 03:

Write a Input Process Output (IPO) to find an area of a circle where

area = pi * radius * radius

Data	Processing	Module	Output
- radius	- Enter radius - area = 3.14 x radius x radius - Display area - end	1000 2000 3000 0000	- Area of a circle

IV. Drawing the Program Flowcharts

- Flowchart is the graphic representations of the individual steps or actions to implement a particular module.
- The flowchart can be likened to the blueprint of a building. An architect draws a blueprint before beginning construction on a building, so the programmer draws a flowchart before writing a program.
- Flowchart is independent of any programming language.
- Flowchart is the logical design of a program.
- It is the basis from which the actual program code is developed.
- Flowchart serves as documentation for computer program.
- The flowchart must be drawn according to definite rules and utilizes standard symbols adopted internationally.

The International Organization for Standardization (IOS) was the symbols shown below (You can draw the symbols using ready-made flowcharting template):

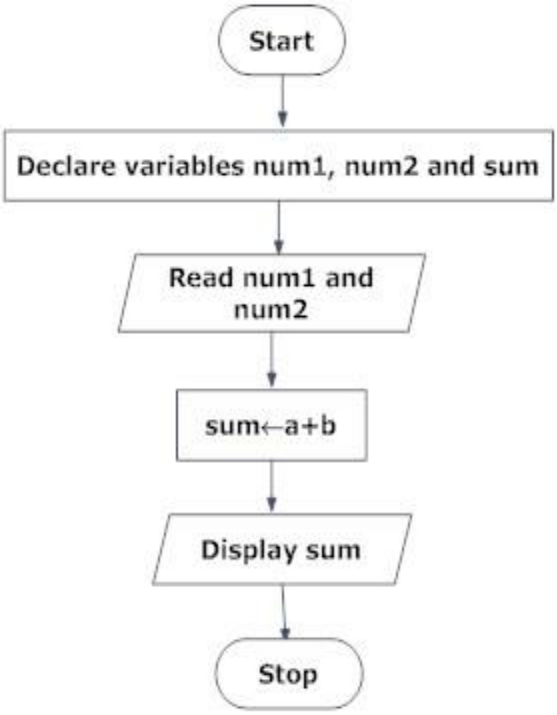
Symbols Used In Flowchart :

Different symbols are used for different states in flowchart, For example: Input/Output and decision making has different symbols. The table below describes all the symbols that are used in making flowchart

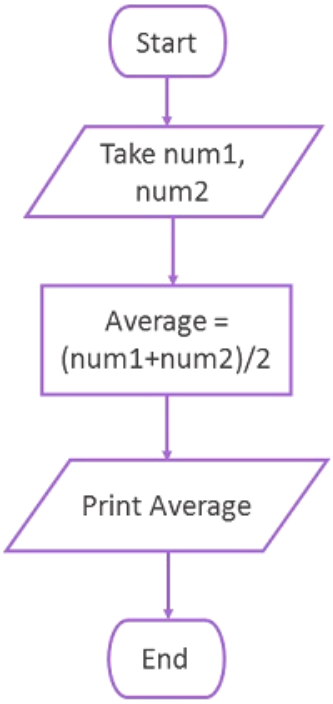
Symbol	Purpose	Description
	Flow line	Used to indicate the flow of logic by connecting symbols.
	Terminal(Stop/Start)	Used to represent start and end of flowchart.
	Input/Output	Used for input and output operation.
	Processing	Used for airthmetic operations and data-manipulations.
	Desicion	Used to represent the operation in which there are two alternatives, true and false.
	On-page Connector	Used to join different flowline
	Off-page Connector	Used to connect flowchart portion on different page.
	Predefined Process/Function	Used to represent a group of statements performing one processing task.

Examples of flowcharts

- Draw a flowchart to add two numbers entered by user.



- Draw a flowchart to calculate the average of two numbers.



- Draw a flowchart to calculate the area of circle..

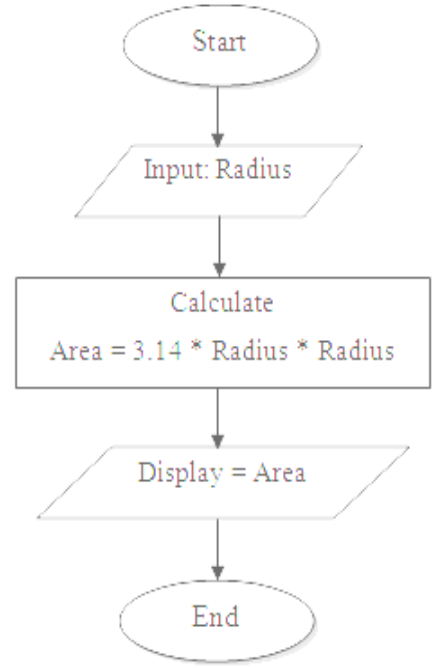


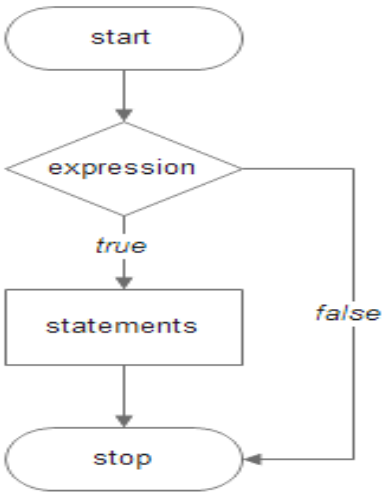
Fig. Flowchart to print Area of Circle

Flowchart of if statement:

The if statement allows you to control the execution of code based on a particular condition. The syntax of the if statement is as follows:

Syntax:

```
if(expression)
{
/* unit of code to be executed */
}
```

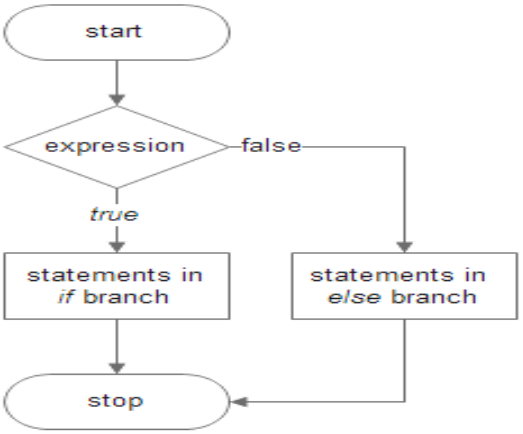


Flowchart of if else statement:

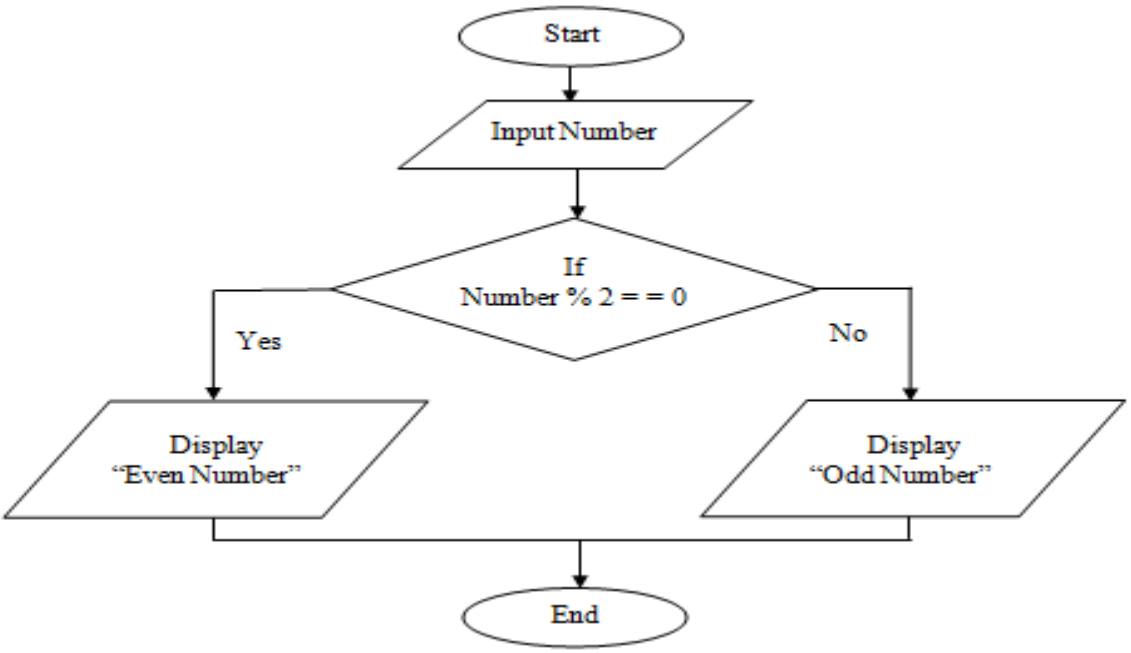
Sometimes you want to execute a piece of code in case of the *expression* in the *if* statement evaluates to *false*. You can use the second form of the if statement which is known as *if else* statement.

Syntax:

```
if(expression)
{
/* code block of if statement*/
}
else
{
/* code block of if statement */
}
```



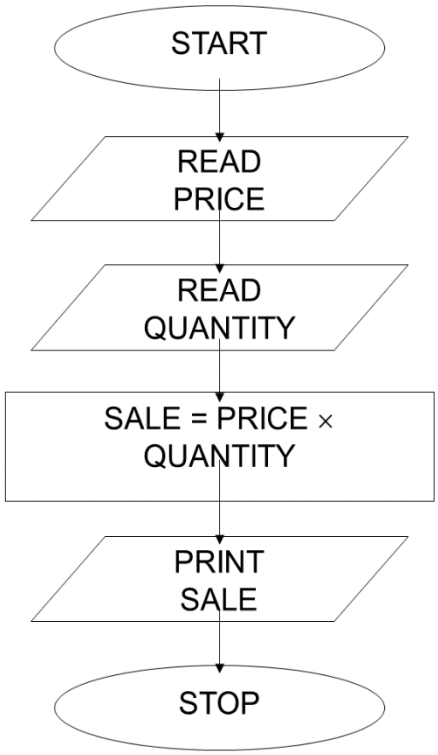
To check whether nmber is even or odd.



V. Writing the Algorithm (Pseudo code)

- Pseudo code means an imitation computer code.
- It is used in place of symbols or a flowchart to describe the logic of a program. Thus, it is a set of instructions (descriptive form) to describe the logic of a program.
- Pseudo code is close to the actual programming language.
- Using the Pseudo code, the programmer can start to write the actual code.

Example # 12:



Algorithm:

Start

Read price, quantity
Sale = price x quantity
Print Sale

End