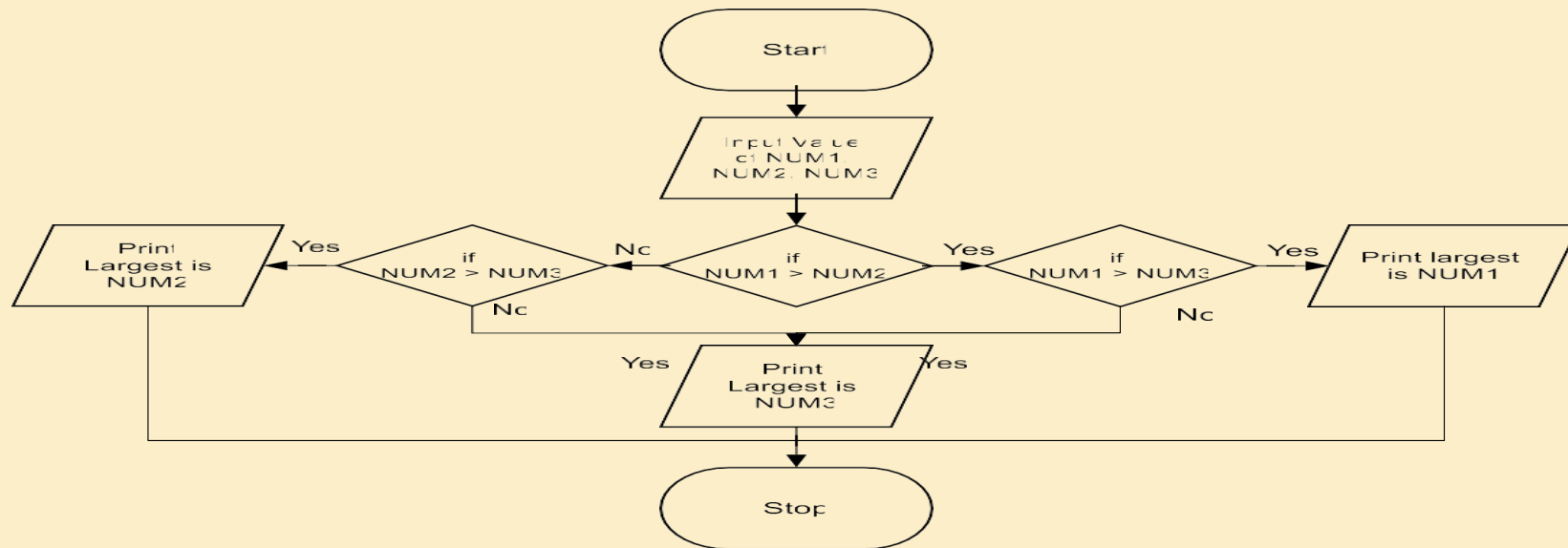


ALGORITHM & FLOWCHART COMP101



Dr. Zaid Ameen

Algorithm & Flowchart

COMP101

Algorithm and flowchart are the powerful tools for learning programming. An algorithm is a step-by-step analysis of the process, while a flowchart explains the steps of a program in a graphical way. Algorithm and flowcharts helps to clarify all the steps for solving the problem. For beginners, it is always recommended to first write algorithm and draw flowchart for solving a problem and then only write the program.

Beginners find it difficult to write algorithm and draw flowchart. The algorithm can vary from person to person to solve a particular problem. The manual will be useful for the students to learn algorithm and flowchart. It includes basics of algorithm and flowchart along with number of examples.

Algorithm & Flowchart

..

ALGORITHM:

The word “algorithm” relates to the name of the mathematician Al-khowarizmi, which means a procedure or a technique. Software Engineer commonly uses an algorithm for planning and solving the problems. An algorithm is a sequence of steps to solve a particular problem or algorithm is an ordered set of unambiguous steps that produces a result and terminates in a finite time

Algorithm has the following characteristics

- **Input:** An algorithm may or may not require input
- **Output:** Each algorithm is expected to produce at least one result
- **Definiteness:** Each instruction must be clear and unambiguous.
- **Finiteness:** If the instructions of an algorithm are executed, the algorithm should terminate after finite number of steps

The algorithm and flowchart include following three types of control structures.

- 1. Sequence:** In the sequence structure, statements are placed one after the other and the execution takes place starting from up to down.
- 2. Branching (Selection):** In branch control, there is a condition and according to a condition, a decision of either TRUE or FALSE is achieved. In the case of TRUE, one of the two branches is explored; but in the case of FALSE condition, the other alternative is taken. Generally, the 'IF-THEN' is used to represent branch control.
- 3. Loop (Repetition):** The Loop or Repetition allows a statement(s) to be executed repeatedly based on certain loop condition e.g. WHILE, FOR loops.

Advantages of algorithm

- It is a step-wise representation of a solution to a given problem, which makes it easy to understand.
- An algorithm uses a definite procedure.
- It is not dependent on any programming language, so it is easy to understand for anyone even without programming knowledge.
- Every step in an algorithm has its own logical sequence so it is easy to debug.

Algorithm & Flowchart

HOW TO WRITE ALGORITHMS

Step 1 **Define your algorithms input:** Many algorithms take in data to be processed, e.g. to calculate the area of rectangle input may be the rectangle height and rectangle width.

Step 2 **Define the variables:** Algorithm's variables allow you to use it for more than one place. We can define two variables for rectangle height and rectangle width as HEIGHT and WIDTH (or H & W). We should use meaningful variable name e.g. instead of using H & W use HEIGHT and WIDTH as variable name.

Step 3 **Outline the algorithm's operations:** Use input variable for computation purpose, e.g. to find area of rectangle multiply the HEIGHT and WIDTH variable and store the value in new variable (say) AREA. An algorithm's operations can take the form of multiple steps and even branch, depending on the value of the input variables.

Step 4 **Output the results of your algorithm's operations:** In case of area of rectangle output will be the value stored in variable AREA. if the input variables described a rectangle with a HEIGHT of 2 and a WIDTH of 3, the algorithm would output the value of 6.

FLOWCHART:

The first design of flowchart goes back to 1945 which was designed by John Von Neumann. Unlike an algorithm, Flowchart uses different symbols to design a solution to a problem. It is another commonly used programming tool. By looking at a Flowchart one can understand the operations and sequence of operations performed in a system. Flowchart is often considered as a blueprint of a design used for solving a specific problem.

Advantages of flowchart:

- Flowchart is an excellent way of communicating the logic of a program.
- Easy and efficient to analyze problem using flowchart.
- During program development cycle, the flowchart plays the role of a blueprint, which makes program development process easier.
- After successful development of a program, it needs continuous timely maintenance during the course of its operation. The flowchart makes program or system maintenance easier.
- It is easy to convert the flowchart into any programming language code.

..

Algorithm & Flowchart to find the sum of two numbers

Algorithm

Step-1 Start

Step-2 Input first numbers say A

Step-3 Input second number say B

Step-4 $SUM = A + B$

Step-5 Display SUM

Step-6 Stop

Algorithm

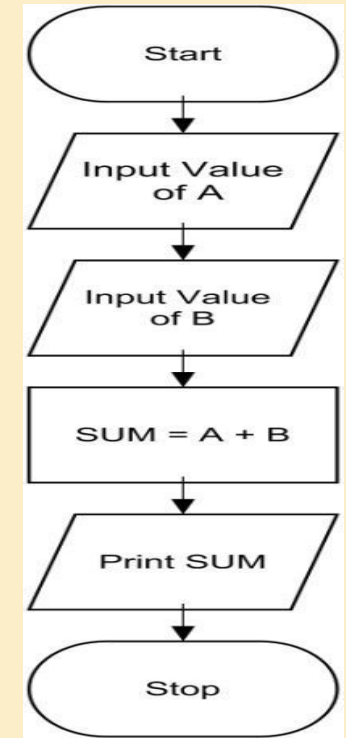
Step-1 Start

Step-2 Input two numbers say A & B

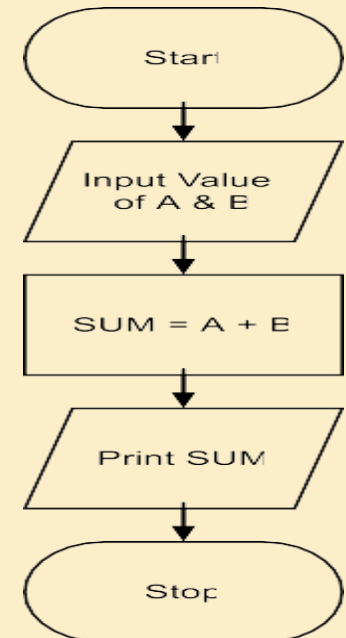
Step-3 $SUM = A + B$

Step-4 Display SUM

Step-5 Stop

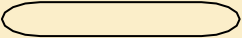
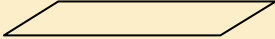
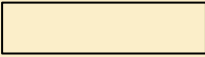
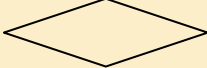




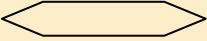
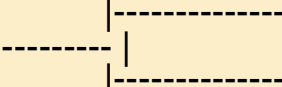


OR



Algorithm & Flowchar

Flowchart is diagrammatic /Graphical representation of sequence of steps to solve a problem. To draw a flowchart following standard symbols are use

Symbol Name	Symbol	function
Oval		Used to represent start and end of flowchart
Parallelogram		Used for input and output operation
Rectangle		Processing: Used for arithmetic operations and data-manipulations
Diamond		Decision making. Used to represent the operation in which there are two/three alternatives, true and false etc
Arrows		Flow line Used to indicate the flow of logic by connecting symbols
Circle		Page Connector
		Off Page Connector
		Predefined Process /Function Used to represent a group of statements performing one processing task.
		Preprocessor
		Comments

Mathematical Operators:

Operator	Meaning	Example
+	Addition	$A + B$
-	Subtraction	$A - B$
*	Multiplication	$A * B$
/	Division	A / B
^	Power	A^3 for A^3
%	Reminder	$A \% B$

Relational Operators

Operator	Meaning	Example
<	Less than	$A < B$
<=	Less than or equal to	$A <= B$
= or ==	Equal to	$A = B$
# or !=	Not equal to	$A \# B$ or $A != B$
>	Greater than	$A > B$
>=	Greater than or equal to	$A >= B$

Algorithm & Flowchart

Algorithm & Flowchart to convert temperature from Celsius to Fahrenheit

C : temperature in Celsius F : temperature Fahrenheit

Algorithm

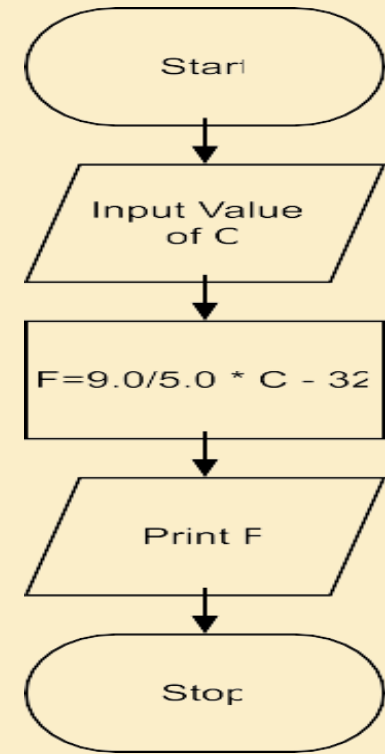
Step-1 Start

Step-2 Input temperature in Celsius say C

Step-3 $F = (9.0/5.0 \times C) + 32$

Step-4 Display Temperature in Fahrenheit F

Step-5 Stop



Algorithm & Flowchart to convert temperature from Fahrenheit to Celsius

C : temperature in Celsius F : temperature Fahrenheit

Algorithm

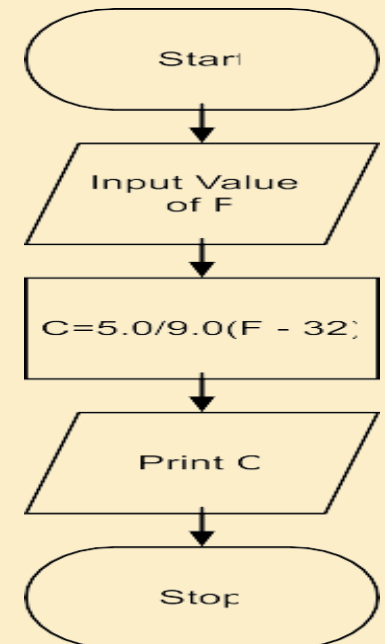
Step-1 Start

Step-2 Input temperature in Fahrenheit say F

Step-3 $C = 5.0/9.0 (F - 32)$

Step-4 Display Temperature in Celsius C

Step-5 Stop



Algorithm & Flowchart

..

Algorithm & Flowchart to find Area and Perimeter of Square

L : Side Length of Square AREA : Area of Square
PERIMETER : Perimeter of Square

Algorithm

Step-1 Start

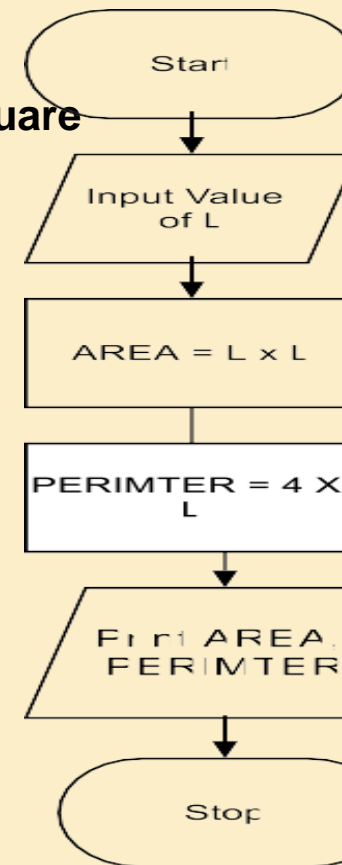
Step-2 Input Side Length of Square say L

Step-3 Area = L x L

Step-4 PERIMETER = 4 x L

Step-5 Display AREA, PERIMETER

Step-6 Stop



Algorithm & Flowchart to find Area and Perimeter of Rectangle

L : Length of Rectangle B : Breadth of Rectangle
AREA : Area of Rectangle
PERIMETER : Perimeter of Rectangle

Algorithm

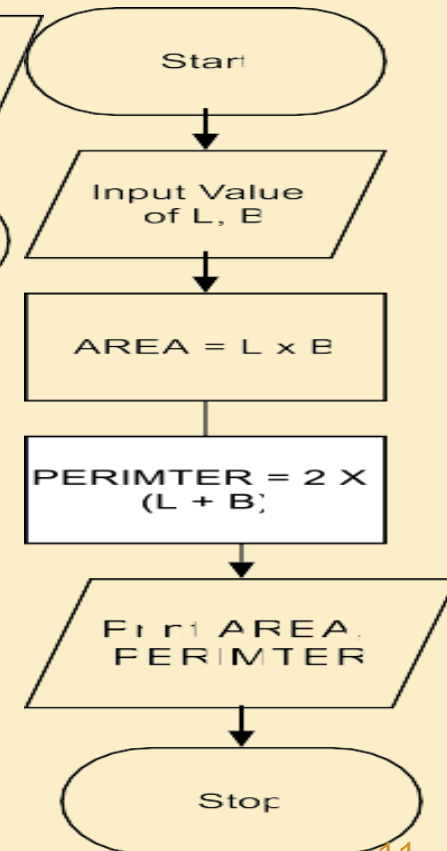
Step-1 Start

Step-2 Input Side Length & Breadth say L, B

Step-3 Area = L x B

Step-4 PERIMETER = 2 x (L + B)

Step-5 Display AREA, PERIMETER Step-6 Stop



Algorithm & Flowchart

Algorithm & Flowchart to find Area and Perimeter of Circle

R : Radius of Circle AREA : Area of Circle
PERIMETER : Perimeter of Circle

Algorithm

Step-1 Start

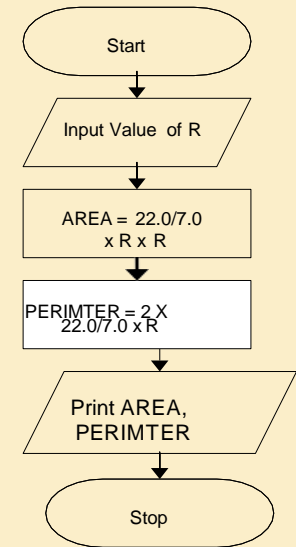
Step-2 Input Radius of Circle say R

Step-3 Area = $22.0/7.0 \times R \times R$

Step-4 PERIMETER = $2 \times 22.0/7.0 \times R$

Step-5 Display AREA, PERIMETER

Step-6 Stop



Algorithm & Flowchart to find Area & Perimeter of Triangle (when three sides are given)

A : First Side of Triangle B : Second Side of Triangle C : Third Side of Triangle
AREA : Area of Triangle
PERIMETER : Perimeter of Triangle

Algorithm

Step-1 Start

Step-2 Input Sides of Triangle A,B,C

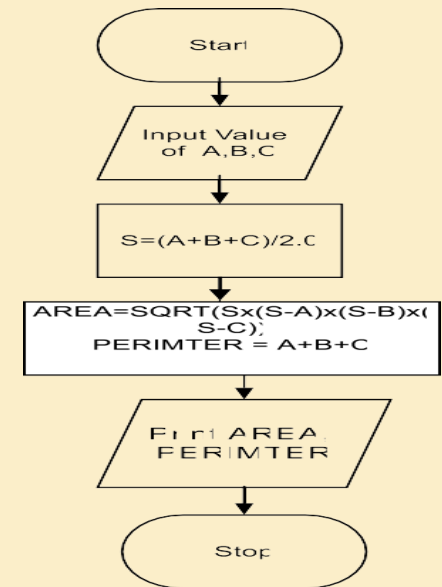
Step-3 $S = (A + B + C) / 2.0$

Step-4 AREA = $\text{SQRT}(S \times (S-A) \times (S-B) \times (S-C))$

Step-5 PERIMETER = A + B + C

Step-6 Display AREA, PERIMETER

Step-7 Stop



Algorithm & Flowchart to find Simple Interest

P : Principle Amount N : Time in Years
 R : % Annual Rate of Interest SI : Simple Interest

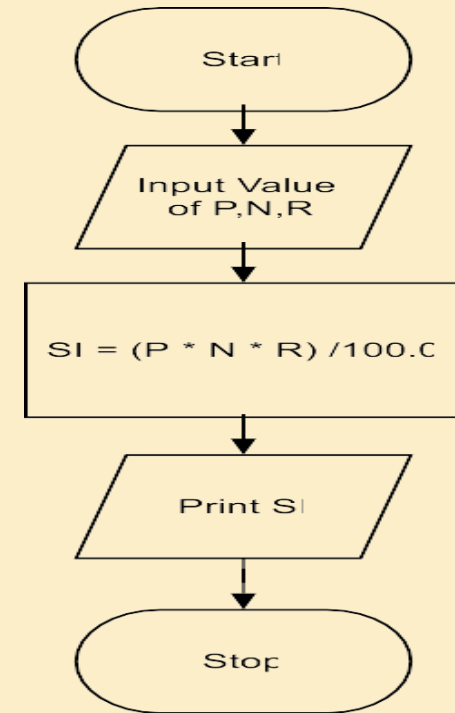
Algorithm

Step-1 Start

Step-2 Input value of P, N, R

Step-3 $SI = (P \times N \times R) / 100.0$

Step-4 Display SI F Step-6 Stop



Algorithm & Flowchart to find Compound Interest

P : Principle Amount N : Time in Years
 R : % Annual Rate of Interest CI : Compound Interest

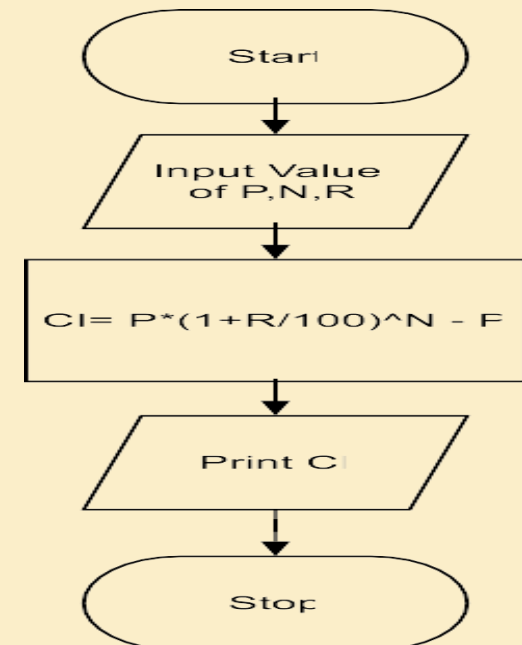
Algorithm

Step-1 Start

Step-2 Input value of P, N, R C

Step-3 $CI = P(1+R/100)^N - P$

Step-4 Display CI Step-6 Stop



..

Algorithm & Flowchart to Swap Two Numbers using Temporary Variable

Algorithm

Step-1 Start

Step-2 Input Two Numbers Say NUM1, NUM2

Step-3 Display Before Swap Values NUM1, NUM2

Step-4 TEMP = NUM1

Step-5 NUM1 = NUM2

Step-6 NUM2 = TEMP

Step-7 Display After Swap Values NUM1, NUM

Step-8 Stop

