ALGORITHM

Algorithm and flowcharts are alternative representations of a program and are considered basic to any programming activity. Without algorithm development, programming activity is considered to be uncompleted.

The dictionary meaning of the word algorithm is "a process or a set of rules used for calculation".

There are various definitions of an algorithm, which are following

1) It can be defined as a prescribed sequence of well-defined and precise instructions for the solution of a given problem.
2) A method that can be used by a computer for the solution of a problem or a sequence of computational steps that transform the input into the output.
3) A step-by-step procedure to solve the given problem is known as Algorithm.
4) It is a sequence of unambiguous instructions for solving a problem.

This algorithm is implemented on a computer system.

Why should we study algorithms?

A good algorithm implemented on a slow computer may perform much better than a bad algorithm implemented on a fast computer.

Every algorithm is supposed to have the following five important features:

1) Definiteness

2) Effectiveness

3) Finiteness

4) Input

5) Output

Definiteness: By definiteness it is implied that each step of the algorithm must specify a definite action i.e., the steps should not be vague. Each step or instruction in an algorithm must be precisely defined. The actions to be carried out must be rigorously and unambiguously specified in each case. For example “Add 2 or 3 to x" is not a valid algorithmic step because it is ambiguous, whereas "Add 2 to x" is a valid. As a result of this statement, the value of x is increased by 2.

Effectiveness: An algorithm is generally expected to be effective. This requires that all the operations to be performed in an algorithm must be sufficiently basic, so that a person using a

pen and pencil must be able to carry them out, in principle, in a finite amount of time. This saves lots of time.

Finiteness: It implies that the algorithm must have finite number of steps. Also the time taken to execute all the steps of the algorithm should be finite and within a reasonable limit. It means that every algorithm must always terminate after the execution of a finite number of steps. Otherwise the procedure is said to enter an infinite loop that never stops until it is forcibly aborted. Such procedures are referred to as computational procedures, for example, operating systems.

Input: Every algorithm takes zero or more inputs. Inputs are the numeric or non-numeric quantities that are given to an algorithm. The algorithm steps use these initially supplied inputs when the algorithm actually executes. In certain algorithm the input is part of algorithm. For example, in numerical algorithm.

Output: An algorithm produce one or more outputs. If there are no output, the algorithm is considered not to have solved any computational problem. These outputs must have specified relations with the corresponding inputs. The output conforms to the inputs. For example, when 2 and 3 are the inputs for a multiplication algorithm, the output should be 6.

Advantages

1) It is simple to understand step by step solution of the problem.
2) It is easy to debug i.e., errors can be easily pointed out.
3) It is independent of programming languages.
4) It is compatible in the sense that each step of algorithm can be easily coded into its equivalent in high level languages.

Method for Developing an Algorithm

1. Define the problem: State the problem you are trying to solve in clear and concise terms.

2. List the inputs (information needed to solve the problem) and the outputs (what the algorithm will produce as a result)

3. Describe the steps needed to convert or manipulate the inputs to produce the outputs. Start at a high level first, and keep refining the steps until they are effectively computable operations.

4. Test the algorithm: choose data sets and verify that your algorithm works.

Example: Finding the largest of three unequal positive numbers

Input: Three unequal positive numbers X, Y and Z

Output: Largest number among X, Y and Z.

i) Start
ii) Read numbers X, Y and Z.
iii) If X is greater than Y, go to step 4. Else go to step 5
iv) If X is greater than Z, go to step 6. Else go to step 8
v) If Y is greater than Z, go to step 7. Else go to step 8
vi) Print X is largest. Go to step 9
vii) Print Y is largest. Go to step 9
viii) Print Z is largest. Go to step 9
ix) Stop

Flow Chart

There are many techniques for developing and representing the program design. One of the most commonly used methods is the use of flow charts. The flowchart graphically represent the logic needed to solve a programming language. A program flowchart represents the detailed sequence of steps, needed to solve the problem.

- The graphical or visual representation of algorithm is called as flow chart.
- It is independent of computer languages.
- The flow charts are easier to understand the flow of the solution.
- Help in algorithm design.
- Check the program logic.
- Help in coding.
- Modification becomes easy.
- Flow charts are drawn with the standard symbols accepted worldwide.
- Each standard flow chart symbol represents on action to be performed such as Start or Stop, input operations Read or Write, decision making etc.
- Better documentation provided.

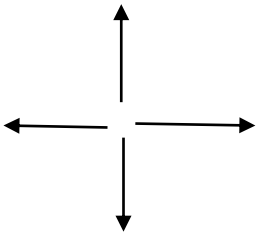
Standard flow chart symbols

1) Terminal (Start or Stop Symbol)

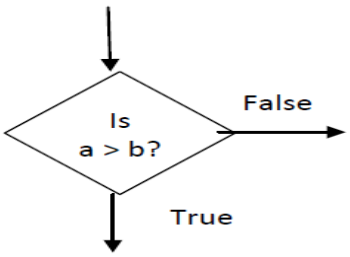
2) Input / Output

3) Processing

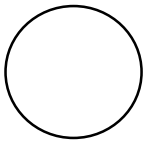
4) Flow Lines: These are arrow mark symbols used to connect two boxes and to indicate the direction of data or processing flow



5) Decision Box: This is a diamond shaped box, which is used to indicate logical checking and gives decision after comparing between two or more objects (Eg. Yes or No; True or False, =, >, <, etc.)

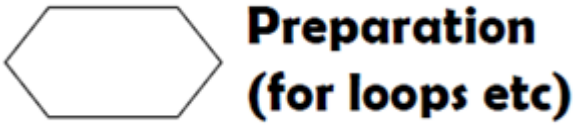


6) Connector: This is a Circular-shaped symbol used to connect different parts of flowchart. When the flow chart is lengthy, it is split into different pages. Then these connectors are used to connect between these pages at the beginning and at the end of each page



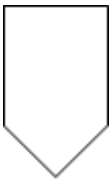
7) Predefined: It is a symbol often used to represent a process that is used several times in the same program. This process is defined only once and reference by this block thereafter.

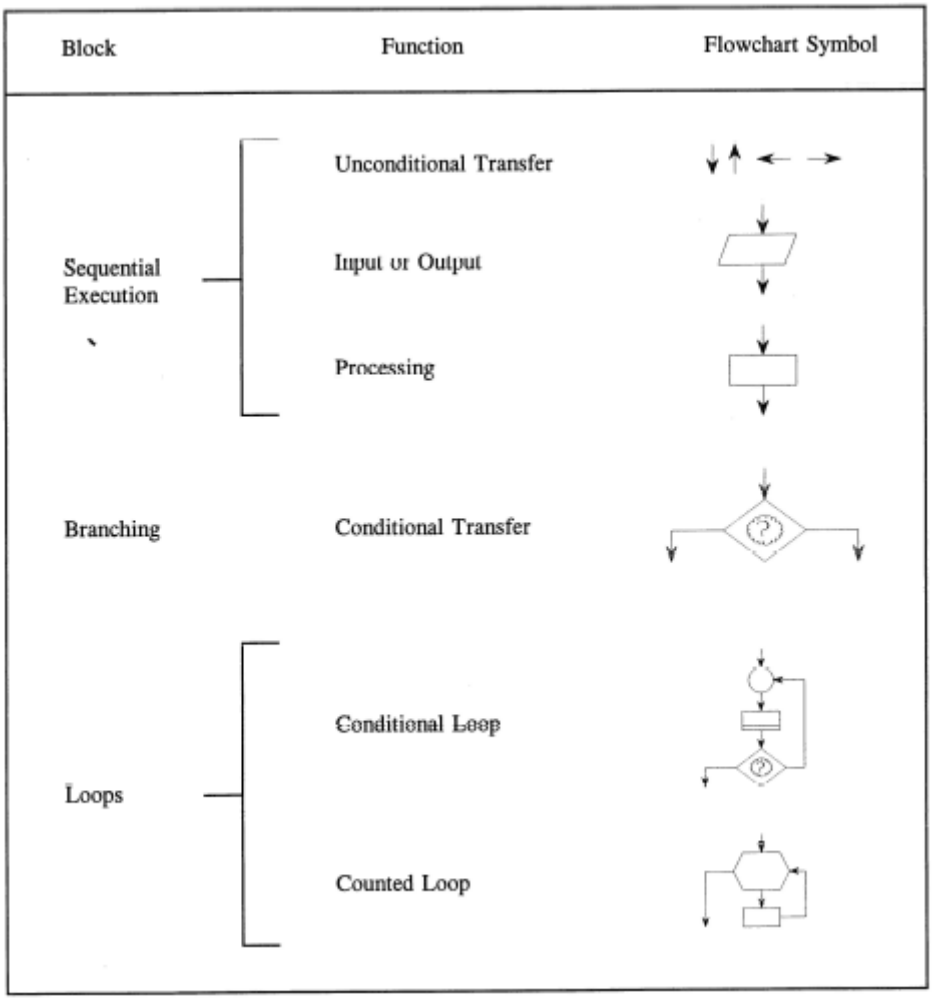
8)



9) Comments

10) Off-pass Connector





General Rules for flowcharting

1. All boxes of the flowchart are connected with Arrows. (Not lines)

2. Flowchart symbols have an entry point on the top of the symbol with no other entry points. The exit point for all flowchart symbols is on the bottom except for the Decision symbol.

3. The Decision symbol has two exit points; these can be on the sides or the bottom and one side.

4. Generally a flowchart will flow from top to bottom. However, an upward flow can be shown as long as it does not exceed 3 symbols.

5. Connectors are used to connect breaks in the flowchart. Examples are:

- From one page to another page.
- From the bottom of the page to the top of the same page.
- An upward flow of more than 3 symbols

6. Subroutines and Interrupt programs have their own and independent flowcharts.

7. All flow charts start with a Terminal or Predefined Process (for interrupt programs or subroutines) symbol.

8. All flowcharts end with a terminal or a contentious loop.

Q) Algorithm and flow chart to find the average of three numbers.

Algorithm

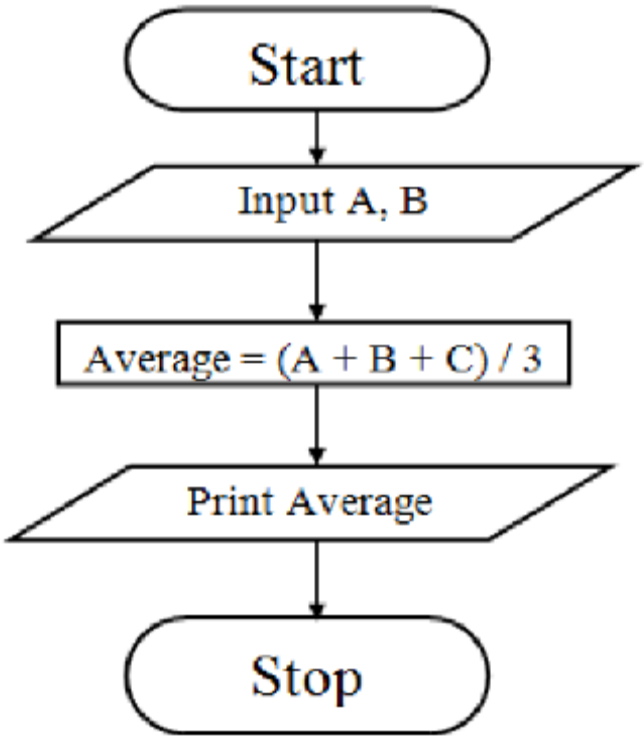
Step1 : Start

Step 2 : Enter Three Numbers A, Band C

Step 3 : Compute Average = (A+B+C)/3

Step 4 : Print Average

Step 5 : Stop



Q) Algorithm and flow chart to find the largest of two numbers.

Algorithm

Step1: Start

Step 2: Enter two numbers A and B

Step 3: Check if A is greater than B if yes go to Step 4 else go to Step 5

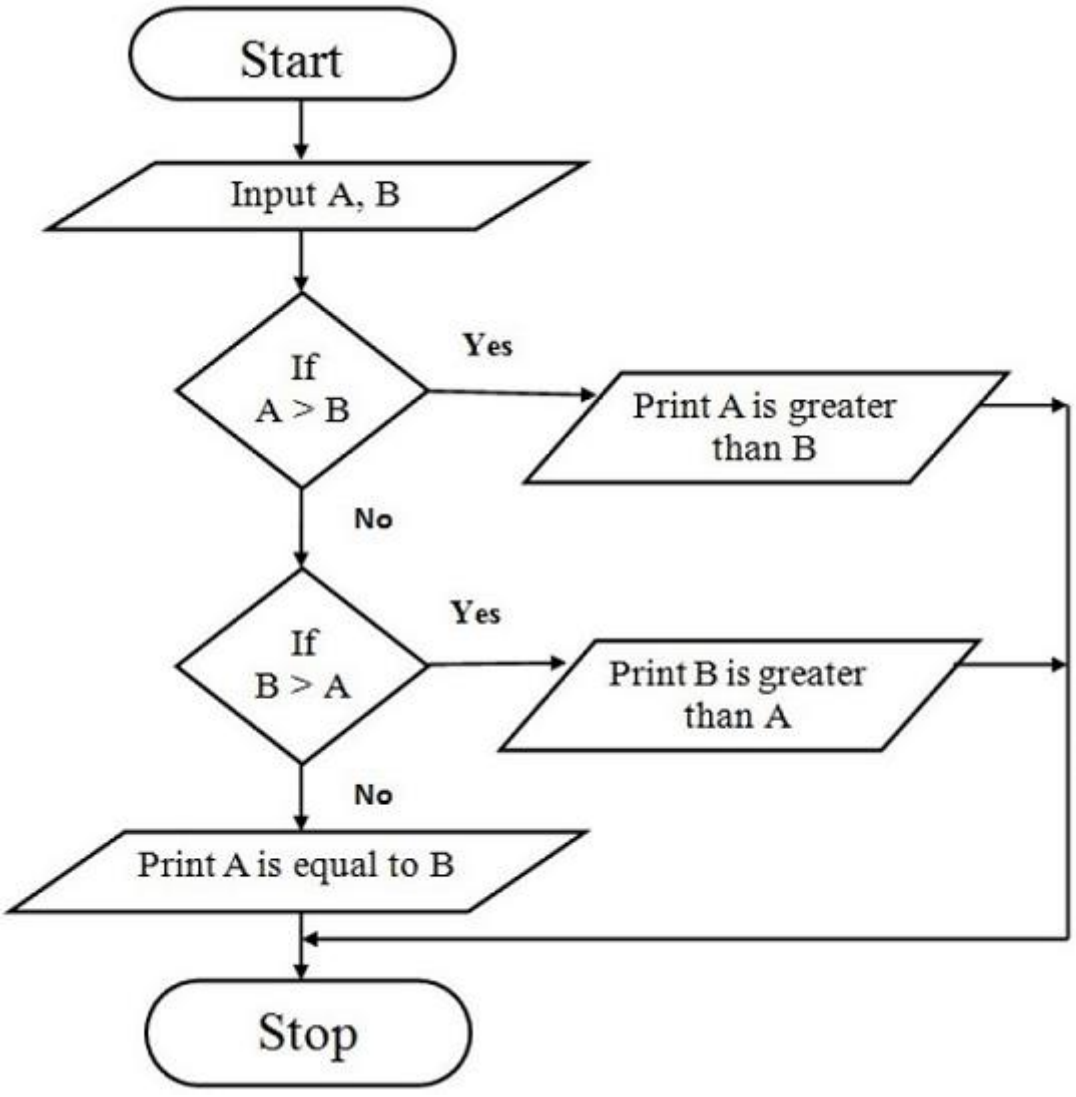
Step 4: Print A is greater than B

Step 5: Check if B is greater than A if yes go to Step 6 else go to Step 7

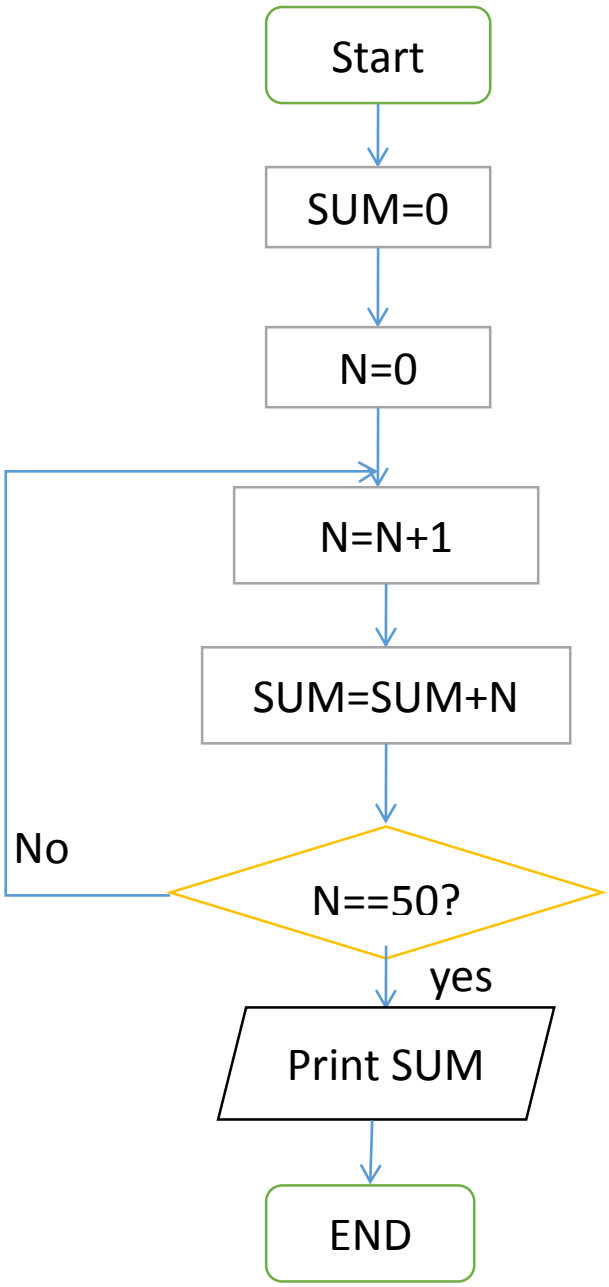
Step 6: Print B is greater than A

Step 7: Print A is equal to B

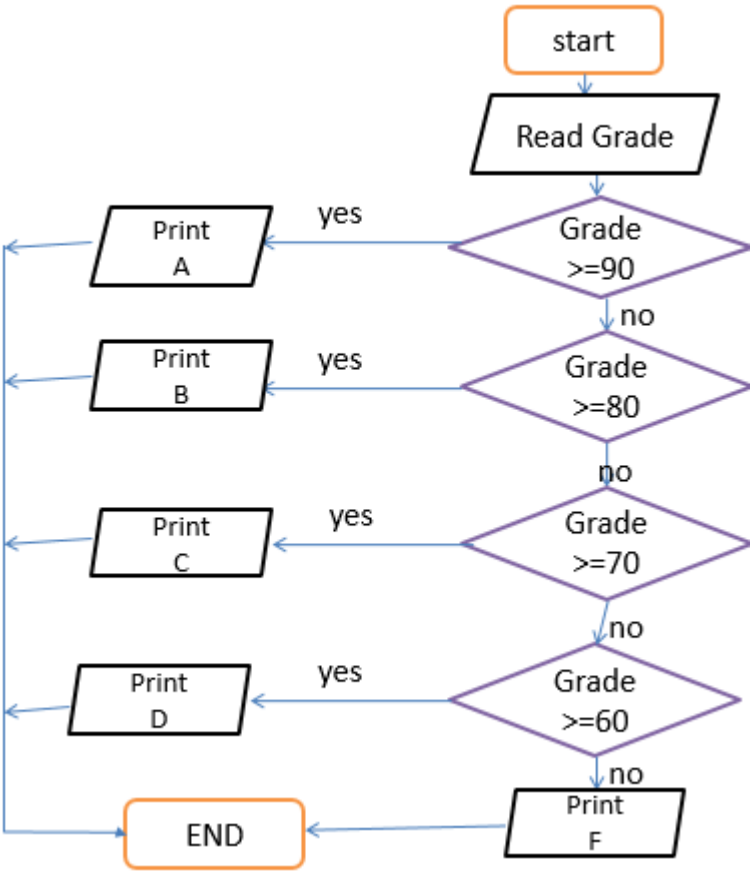
Step 8: Stop



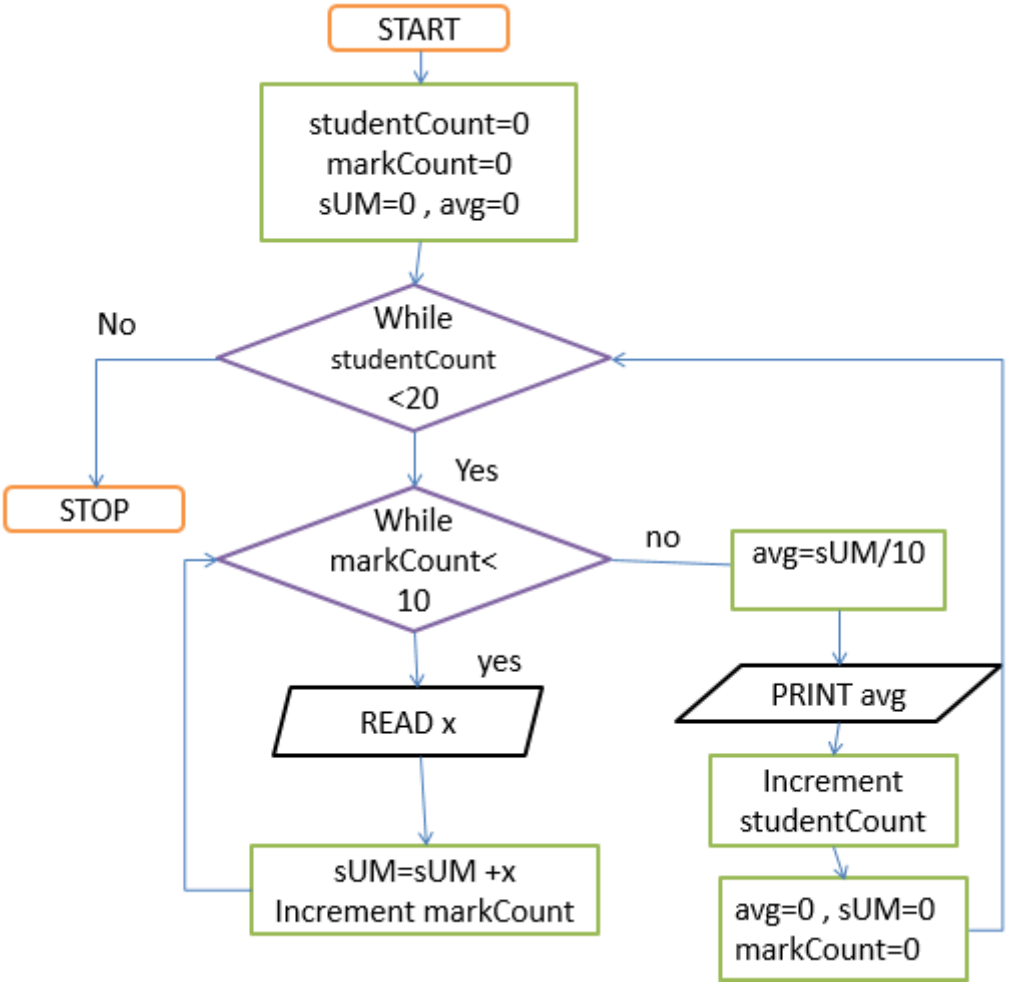
Q) Draw a flowchart to find the sum of first 50 natural numbers

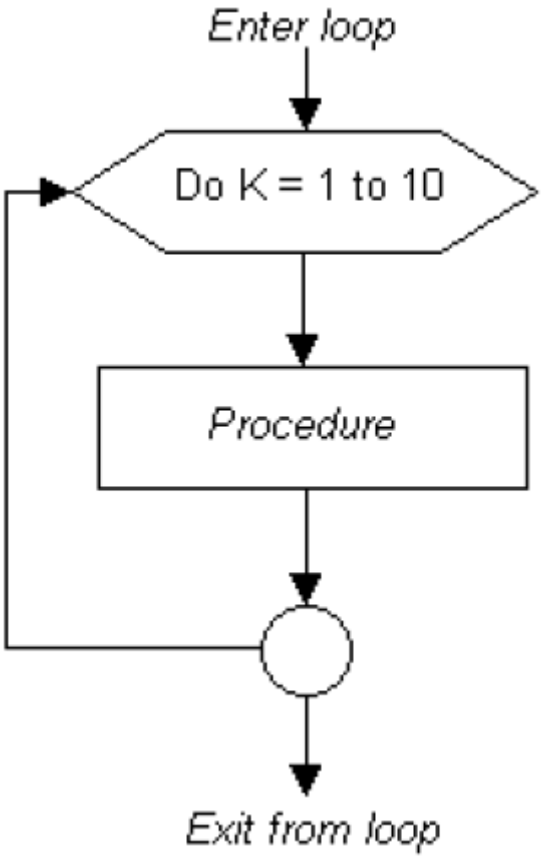


Q) Draw a flowchart for program Calculate the student grade (A,B,C,D,F)



Q) Draw a flowchart for program read (10) marks for (20) students, then print the average for each student





Types of Flowchart

There are three types of flowchart, which are following:-

1) System flowchart
2) Modular flowchart
3) Detail flowchart

Level of Flowchart

There are two level of flowcharts:

1) Macro flowchart: It shows the main segment of a program and shows lesser details
2) Micro flowchart: It shows more detail of any part of the flowchart