**ASSIGNMENT 1**

**1. What Is An Algorithm?**

A **programming algorithm** is a computer procedure that is a lot like a recipe (called a procedure) and tells your computer precisely what steps to take to solve a problem or reach a goal. The ingredients are called inputs, while the results are called the outputs.

In programming, algorithm is the set of well defined instruction in sequence to solve a program. An algorithm should always have a clear stopping point.

**Qualities of a good algorithm**

1. Inputs and outputs should be defined precisely.
2. Each step in algorithm should be clear and unambiguous.
3. Algorithm should be most effective among many different ways to solve a problem.
4. An algorithm shouldn't have computer code. Instead, the algorithm should be written in such a way that, it can be used in similar programming languages.

**Examples Of Algorithms In Programming**

**Write an algorithm to add two numbers entered by user.**

Step 1: Start

Step 2: Declare variables num1, num2 and sum.

Step 3: Read values num1 and num2.

Step 4: Add num1 and num2 and assign the result to sum.

sum←num1+num2

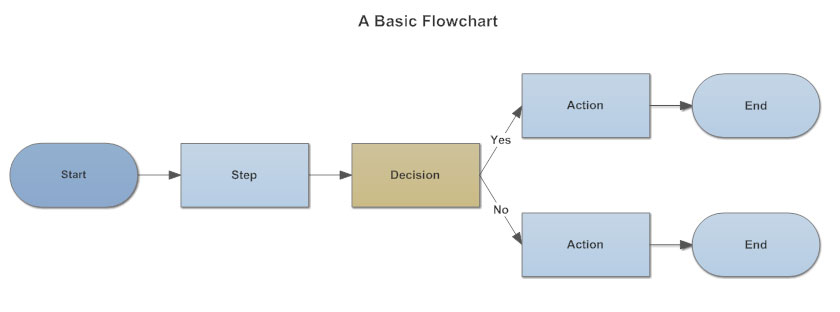
Step 5: Display sum

Step 6: Stop

2**. What is Flowchart?**

A **flowchart** is a type of diagram that represents an algorithm, workflow or process, showing the steps as boxes of various kinds, and their order by connecting them with arrows. This diagrammatic representation illustrates a solution model to a given problem. Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields.

This allows anyone to view the flowchart and logically follow the process from beginning to end.



**3. What are the basic rules to be followed while creating flowchart?**

1. **Proper Form is Essential:** In drawing a proper flowchart, all necessary requirements should be listed out in logical order.

2. **Clarity is Paramount:**The flowchart should be clear, neat and easy to follow. There should not be any room for ambiguity in understanding the flowchart.

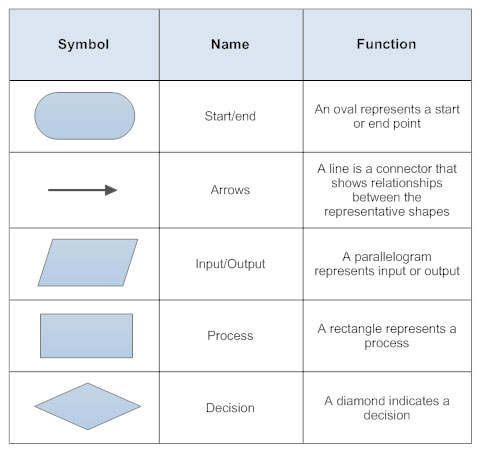
3. **Stick to the Right Direction:**The usual direction of the flow of a procedure or system is from left to right or top to bottom.

4. **Standard for Flow Lines:**Ideally just one flow line should come out from a process symbol.  While only one flow line should enter a decision symbol, around three flow lines (depending on the answer) should leave the decision symbol. Additionally, only one flow line is utilized together with a terminal symbol.

5. **Be Concise, not Copious:**Write within standard symbols briefly.

6. **Logic precedes everything:**If you are dealing with a complex flowchart then use connector symbols to minimize the number of flow lines. Ditch the intersection of flow lines to ensure effectiveness and better communication. It is imperative that your flowchart has a logical start and finish.

**4. Different Symbols used for creating flowchart?**



5. Condition and looping and how they are helpful in applying the logic?

In computer programming, **conditional loops** or **repetitive control structures** are a way for computer programs to repeat one or more various steps depending on conditions set either by the programmer initially or real-time by the actual program.

A conditional loop has the potential to become an infinite loop when nothing in the loop's body can affect the outcome of the loop's conditional statement. However, infinite loops can sometimes be used purposely, often with an exit from the loop built into the loop implementation for every computer language, but many share the same basic structure and/or concept. The While loop and the For loop are the two most common types of conditional loops in most programming languages.