

**Rajmani Kumar,**  
**Lecturer, Dept. of BCA**  
**S.U.College, Hilsa (Nalanda)**  
**Patliputra University, Patna**

**BCA-1<sup>st</sup> Year**

**Paper-I**

## **Programming Concepts**

### **Programming definition**

A computer program is a sequence of instructions written using a Computer Programming Language to perform a specified task by the computer.

- A computer program is also called a computer software, which can range from two lines to millions of lines of instructions.
- Computer program instructions are also called program source code and computer programming is also called program coding.
- A computer without a computer program is just a dump box; it is programs that make computers active.

The two important terms that we have used in the above definition are:

- Sequence of instructions
- Computer Programming Language

A program is a set of instruction written in a language (such as BASIC) understandable by the computer to perform a particular function on the computer. A well written program could be parceled well to form an application package customized for solving specific type of problem on the computer system.

A computer programmer is computer scientist (a professional) skilled in using constructs of programming languages to develop executable and acceptable computer programs. A software developer is a programmer. Programmers often work hand in hand with system analysts on large projects.

Programming languages are artificial notational languages created or developed to be used in preparing coded instructions on the computer for later execution by the computer. They are usually composed of series of usage rules (syntax) that determine the meaning (semantics) of expressions written in the language. Each programming language comes handy with its own translator i.e interpreter or compiler as the case may be.

Programming is the art of developing computer programs with the aid of selected programming language by a computer programmer. It is a special skill whose quality is tested by the quality of the resulting program or software. In programming, programming stages must be properly followed.

## Characteristics of good program

### Portability

A program should be supported by many different computers. The program should compile and run smoothly on different platforms. So, portability is measured by how a software application can be transferred from one computer environment to another without failure. A program is said to be more portable, if it is easily adopted in different computer systems.

### Maintainability

It is the process of fixing program errors and improving the program. If a program is easy to read and understand, then its maintenance will be easier. It should also prevent unwanted works, so that the maintenance cost in future will be low. It should also have quality to easily meet new requirements.

### Efficient

Program is said to be more efficient, if it takes less space and easily converted to machine language. The algorithm should be more effective. The program efficiency is also high if it has high speed during runtime execution of program.

### Reliable

The user's actual needs will change from time-to-time, so program is said to be reliable if it works smoothly in every version. It is measured as high reliable if it gives same performance in all simple to complex conditions.

### Machine Independence

Program should be machine independent. Program written on one system should be able to execute on any other without any changes. It is not system specific and provide more flexibility.

### Cost Effectiveness

Cost Effectiveness is the key to measure the program quality. Cost must be measured over the life of the program and must include both cost and human cost of producing these programs.

## **Flexible**

Program should be written in such a manner that it allows to add new features without changing the existing module. It should be always ready to meet new requirements. A high flexible software is always ready for a new world of possibilities.

## **Programming Steps**

Before going on to discuss the five steps of creating a program it is important to determine what exactly a program is. A program is a list of instructions that contain data for a computer to follow. Different programs are written with different languages. An editing program is made with a different programming “language” than one that uses graphics. Some well known programming languages are COBOL -business, BASIC -language, and – C which is used in science. Programming is complicated process, below is general overview of this 5 step procedure.

Contents

- 1 Clarify Programming Needs
- 2 Design the Program
- 3 Code the Program
- 4 Test the Program
- 5 Document and Maintain

## **Clarify Programming Needs**

Knowing the objective is the first consideration. Is it a payroll or editing program? Knowing who the end user will be is also important. Determining the inputs and outputs is next. How will the program operate and what data is needed to make it happen. After this has been decided feasibility is the next consideration. How many programmers will it take, is the project within budget, does the project have a realistic outline. Finally, if the project is a go, then one must take measures to ensure the project is properly documented and analyzed.

Four mini steps:

- Clarify desired processing
- Double – check feasibility of implementing the program
- Document the analysis
- kopp

## **Design the Program**

Programs use algorithms which are like equations that tell the computer what task to perform. The aim of the programmer is to create algorithms that are

clear and simple. Algorithms are expressed first in logical hierarchical form known as modularization. Using modules or (a complete thought) the programmer creates a logical thought process for the computer to follow. After that the program is broken down in greater detail using pseudo code. Pseudo code uses terms like if, else, and, then to relate the programs rules to the computer.

Two mini steps:

1. Determine program logic through top down approach and modularization, using a hierarchy chart
2. Design details using pseudo code and/or flowcharts, preferably involving control structure.

A module, a processing step of a program, made up of logically related program statements.

A hierarchy chart, which represents top-down program design, explains the main purpose of the program.

Pseudo code, a way of designing a program which uses normal language statements in order to describe the logic and the processing flow.

Program flowcharts, graphically shows the detailed series of steps

- Three control structure: Edit
- Sequence control structure – No decision making
- Selection control structure – Loop control structure/Repetition or Iteration Structure

### **Code the Program**

After the program has been designed it must be coded or written. Using the pseudo code and logic requirements from step two an appropriate programming language must be selected. As stated in the introduction, coding languages differ in specifications and usability. Once the appropriate code language has been chosen, it is imperative that the programmer follow the syntax rules with as little deviation as possible in order for the program to have high accuracy.

Two mini steps:

- Select the appropriate high-level programming language
- Code the program in that language following the syntax carefully

### **Test the Program**

After the program is written it then enters the programming debugging and testing phase of the Program Development Life Cycle (PDLC). During this time the programmer will be looking for errors in both logic and syntax, as well as exploring other areas that may cause the program to either not work properly or to not run at all.

This process is a lengthy and tedious one, oftentimes consisting of up to 50% of a program's time in development. However, with a careful eye paid to program design and coding the amount of time spent debugging can be cut considerably.

As stated, debugging will uncover errors in both logic and syntax. Syntax errors will prevent the program from executing. They can be such simple things as misspelled words or can involve breaking the syntax rules of the programming language used.

On the other hand, logic errors will allow the program to run but will provide incorrect results. Errors of this kind may consist of merely using the wrong relational operator or other, larger, mistakes in writing formulas.

Once the programmer locates the errors they are then fixed and the program is run again. This will happen multiple times, often called "execute, check, and correct", until the program runs flawlessly.

The program will then enter the testing phase.

The Bug hard at work inputting errors.

Testing the program comes in two phases, alpha and beta.

Alpha testing is the process of reading through the program in search of errors in logic. The second step is to run a diagnostic program to search for syntax or input errors.

Beta testing involves using the program in the real world to see if it contains any bugs or other deficiencies.

### **Document and Maintain**

Documentation should be ongoing from the very beginning because it is needed for those involved with program now and future. Upon completion User Documentation for commercial use, Operator Documentation for people who run computer systems, and Programmer Documentation for programmers charged with maintenance.

Four mini steps:

1. Write user documentation
2. Write operator documentation
3. Write programmer documentation
4. Maintain the program

## ALGORITHM

Algorithm is a finite sequence of instructions, each of which has a clear meaning and can be performed with a finite amount of effort in a finite length of time. No matter what the input values may be, an algorithm terminates after executing a finite number of instructions.

We represent an algorithm using a pseudo language that is a combination of the constructs of a programming language together with informal English statements.

The ordered set of instructions required to solve a problem is known as an algorithm.

The characteristics of a good algorithm are:

- Precision – the steps are precisely stated (defined).
- Uniqueness – results of each step are uniquely defined and only depend on the input and the result of the preceding steps.
- Finiteness – the algorithm stops after a finite number of instructions are executed.
- Input – the algorithm receives input.
- Output – the algorithm produces output.
- Generality – the algorithm applies to a set of inputs.

### Example

Q. Write a algorithm to find out number is odd or even?

Ans.

```
step 1 : start
step 2 : input number
step 3 : rem=number mod 2
step 4 : if rem=0 then
    print "number even"
    else
    print "number odd"
    endif
step 5 : stop
```

Q. Write an algorithm to add 3 numbers.

Ans.

```
Step1: start
Step2: Input a,b,c
Step3: s=a+b+c
Step4: output s
Step5: stop
```

## FLOWCHART

Flowchart is a diagrammatic representation of an algorithm. Flowchart is very helpful in writing program and explaining program to others.

### Symbols Used In Flowchart

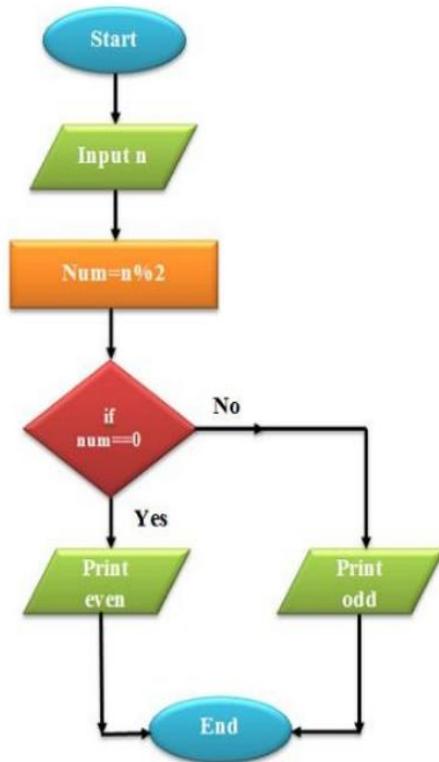
Different symbols are used for different states in flowchart, For example: Input/Output and decision making has different symbols. The table below describes all the symbols that are used in making flowchart

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 arithmetic operations and data manipulations.
	Decision	Used to represent the operation in which there are two alternatives, true and false.
	On-page Connector	Used to join different flow line
	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.

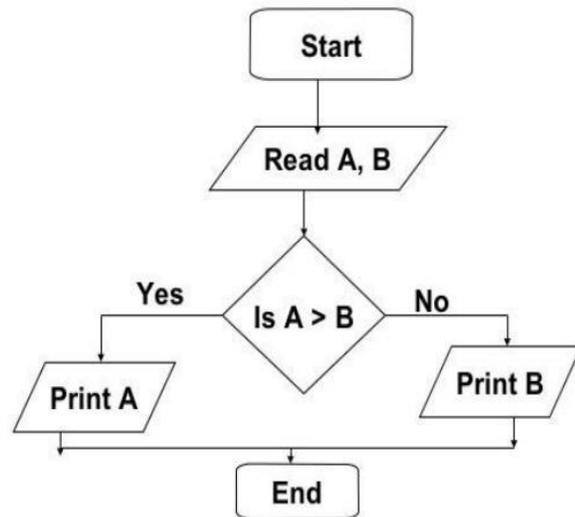
## Examples of flowcharts in programming

Draw a flowchart to odd and even number entered by user

Flow Chart for ODD and Even Number

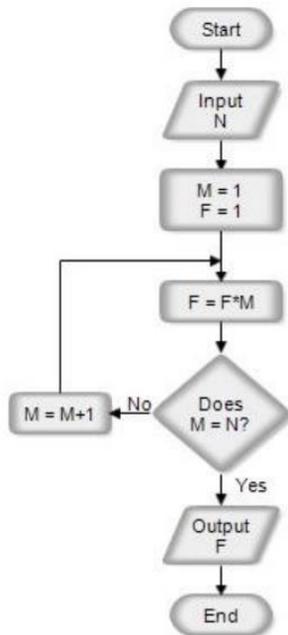


Flow Chart to find largest of two numbers:



Flowchart for find the Factorial of N Numbers and sum & average of 3 Numbers

Flowchart to find Factorial of N



Flowchart to find sum and average of 3 numbers.

