

## Lecture Note 01

Today it is very hard to find a person who has not used the computer by any means or almost impossible to find someone who has not seen usage of computer, therefore, it is convenient to define **programming** in those terms. Everything you see on the computer has programming on the backend. Like using **browser** for surfing, reading a document using **Adobe**, editing on **Word**, making spread sheets using **Excel**, creating slides using **Power point**, playing **games**, listening **audios**, watching **videos**, using **customized** programs like Attendance System, Inventory System, Sales System etc. Programming essentially involves many steps from problem definition to deployment (these phases will be discussed later). We may say that there is some **program** behind everything going on computer and **programming** is art of developing programs; whereas; program is just set of statement(s). Programs may have 1 statement or any number of statements. Typically a serious program has thousand lines of code; whereas; it is not hard to find a program having more than 1 lac statements. One lac statements may be surprising for beginners but just imagine a big building and try to estimate number of bricks used to build that building. Essentially programming is not very different from constructing a building, as mason has to choose each brick and place it according to requirements, similarly a programmer (One who creates programs) has to choose every statement and place it according to requirements.

Technically a program is set of small steps required to perform a task. Apparently it may seem too simple but just consider steps to make a cup of tea:

1. put water into a pan
2. on stove (bring fire to stove)
3. put pan over stove
4. when water start boiling:
  - a. add tea leaves into pan
5. add sugar
6. add milk
7. make 1 or 2 boils
8. pour tea into cup

Similarly steps may be different if you are making tea from tea bag and you have facility of a dispenser having hot water:

1. take hot water into cup
2. put tea bag into cup
3. add sugar
4. add milk
5. stir all

Motivation for making tea is a desire to take tea. Similarly a program starts from some problem require programming. Like calculating:

- Given 3 sides of triangle find perimeter
- Given width and length find area of rectangle
- Given heights of students find average height
- Given sale of last year predict sale for current year
- Last one interesting for most of you how much run rate is required to qualify for semi-final?

A program typically has set of statements each of them executed step by step in sequence. Each statement is executed on its turn. Order of their execution does matter. For example if you write statements:

```
- print "*****"
- print "W E L C O M E"
- print "*****"
```

Obviously result of these statements will be Welcome message covered by stars but if we interchange statement 2 & 3 the result will be stars followed by welcome which is different from required one. Similarly there are program having input from user. For example program to calculate square root of a number requires number as input. Now input is required before calculation like:

```
- input n
- calculate square root of n
- print square root
```

Obviously if we change above sequence the result will be wrong. Apart from sequence of statements some programs required **conditional** statements that may or may not execute depending on condition. For example program to show whether number is even or odd?

```
- input n
- if n is divisible by 2
  ▪ print "EVEN"
- else
  ▪ print "ODD"
```

Now in above set of statements print "EVEN" and print "ODD" are two statements which are depending on whether n is divisible or not. If n is divisible then print "EVEN" statement will execute and otherwise print "ODD".

Lastly there is a **repetition** means to repeat set of statements number of times like:

```
- t=1
```

```
- while t<=10
  ▪ print "Pakistan Zindabad"
  ▪ add 1 to t
```

Above program will print "Pakistan Zindabad" ten times though print statement is written one time but using repetition mechanism it will be executed ten times. Normally every program has all 3 ingredients, **sequential statements**, **conditional statements** and **repetition statements**. Students will gradually learn them step by step. Apart from these it is hard to find programs missing calculations. Therefore, student should spend time to understand how to transform mathematical formulas into programming statements. Typically hard part is conformity that is to check whether they succeed to correctly transform formulas.

### Summary

Computers and electronic gadgets (smart mobiles, tablets etc.) are now a days in every walk of life and as we said before that everything going on these devices has a program or set of programs working behind. As usage of these devices is increasing day by day as is the need of programs and ultimately jobs for programmers.

Essentially a program is set of statements to be executed step by step. Additionally programs may have conditional and repetition statements according to the requirements of programs. Essentially a program initiates from an existing problem, followed by set of procedures [requirements gathering, designing, coding, testing etc.] which ultimately leads towards the completion and deployment of program.