LECTURE NOTE-2

INTRODUCTION TO PROGRAMMING

A language that is acceptable to a computer system is called a ***computer language*** or ***programming language*** and the process of creating a sequence of instructions in such a language is called ***programming*** or ***coding***. A program is a set of instructions, written to perform a specific task by the computer. A set of large program is called ***software***. To develop software, one must have knowledge of a programming language.

Before moving on to any programming language, it is important to know about the various types of languages used by the computer. Let us first know what the basic requirements of the programmers were & what difficulties they faced while programming in that language.

COMPUTER LANGUAGES

Languages are a means of communication. Normally people interact with each other through a language. On the same pattern, communication with computers is carried out through a language. This language is understood both by the user and the machine. Just as every language like English, Hindi has its own grammatical rules; every computer language is also bounded by rules known as *syntax* of that language. The user is bound by that syntax while communicating with the computer system.

Computer languages are broadly classified as:

- ***Low Level Language***: The term low level highlights the fact that it is closer to a language which the machine understands.

  The low level languages are classified as:

  - ***Machine Language***: This is the language (in the form of 0's and 1's, called binary numbers) understood directly by the computer. It is machine dependent. It is difficult to learn and even more difficult to write programs.

  - ***Assembly Language***: This is the language where the machine codes comprising of 0'sand 1's are substituted by symbolic codes (called mnemonics) to improve their understanding. It is the first step to improve programming structure. Assembly language programming is simpler and less time consuming than machine level programming, it is easier to locate and correct errors in assembly language than in machine language programs. It is also machine dependent. Programmers must have knowledge of the machine on which the program will run.

- ***High Level Language:*** Low level language requires extensive knowledge of the hardware since it is machine dependent. To overcome this limitation, high level language has been evolved which uses normal English, which is easy to understand to solve any problem. High level languages are computer independent and programming becomes quite easy and simple. Various high level languages are given below:

    - BASIC (Beginners All Purpose Symbolic Instruction Code): It is widely used, easy to learn general purpose language. Mainly used in microcomputers in earlier days.
    - COBOL (Common Business Oriented language): A standardized language used for commercial applications.
    - FORTRAN (Formula Translation): Developed for solving mathematical and scientific problems. One of the most popular languages among scientific community.
    - C: Structured Programming Language used for all purpose such as scientific application, commercial application, developing games etc.
    - C++: Popular object oriented programming language, used for general purpose.

PROGRAMMING LANGUAGE TRANSLATORS

As you know that high level language is machine independent and assembly language though it is machine dependent yet mnemonics that are being used to represent instructions are not directly understandable by the machine. Hence to make the machine understand the instructions provided by both the languages, programming language instructors are used. They transform the instruction prepared by programmers into a form which can be interpreted & executed by the computer. Flowing are the various tools to achieve this purpose:

- ***Compiler***: The software that reads a program written in high level language and translates it into an equivalent program in machine language is called as compiler. The program written by the programmer in high level language is called source program and the program generated by the compiler after translation is called as object program.

- ***Interpreter***: it also executes instructions written in a high level language. Both complier & interpreter have the same goal i.e. to convert high level language into binary instructions, but their method of execution is different. The complier converts the entire source code into machine level program, while the interpreter takes 1 statement, translates it, executes it & then again takes the next statement.

- ***Assembler***: The software that reads a program written in assembly language and translates it into an equivalent program in machine language is called as assembler.