Programming Languages

- A programming language is an artificial language designed to communicate instructions to a computer
- A programming language is a notation for writing programs.
- A vocabulary and set of grammatical rules for instructing a computer to perform specific tasks.
- Each language has a unique set of keywords (special words that it understands) and a special syntax (format) for organizing program instructions.

There are many programming languages. For example:

GW Basic, C, C++, Java, Pascal, COBOL, Python, C#, PERL, PHP, Visual Basic, etc.

Types of Programming Language

Low–Level Languages

Middle–Level Languages

High–Level Languages

Low–Level Language

- A low level language is one which is closer to the machine (computer).
- It is easier for machines to understand and difficult for humans to understand.
- It is faster in execution as compared to high and middle level languages.

Two of the types of low level languages are:

Machine Language

Assembly Language

Machine Language

- It is one of the low level language.
- It is the language of 0s and 1s.
- Machine languages are the only languages directly understood by the computers.
- While easily understood by computers, machine languages are almost impossible for humans to use because they consist entirely of numbers (0s and 1s).
- It is the native language of the machines (computers).

Here all the instructions are written as code of binary sequence. For example:

- In order to do addition, the code is: **10010001**
- In order to decrement a number by one, the code is: **11011011**

- In order to move data from one place to another, the code is: **10000111**

Assembly Language

- Assembly language is same as machine language but uses English like words to represent individual operations.
- For example: Instead of binary codes it uses: ADD, MOV, SUB, INC
- Assembly language is also a low-level language.
- It is easier than the machine language but still it is very difficult to control a larger program using assembly.
- As assembly language contains English like words, which will not be understood by the computer (because it only understands 0s and 1s)
- A translator first converts the assembly language program into machine language program.
- Translator used with assembly language is called Assembler.

Assembly language program example:

- MOV AH, 02H
- MOV DL, 41H
- INT 21H

Middle-Level Language

- A middle level language is one which is closer to machine (computer) as well as to human (programmer).
- A language that has the features of both low level and high level languages.
- More formally, a high level language that allows you to write low level programs in it is called as middle level language.

Some of the middle level programming languages are:

- C
- IBM PL/S (Programming Language/Systems)
- BCPL (Basic Combined Programming Language)
- BLISS (Bill's Language for Implementing System Software)

High-Level Language

- A high level language is one which is closer to the human (programmer).
- It is easier for humans to understand and difficult for machines to understand.
- It is slower in execution as compared to low level languages.
- Like assembly language, it also uses English like words for the operations.

- For example: for, if, else, break, continue, while, include, using, import
- It is easier than assembly language.

Some of the high level programming languages are:

- GW Basic, C++, JAVA, Pascal, COBOL, Python, C#, Visual Basic, J#, Ruby, PHP

Program example (C# codes):

```
static void Main()
{
    int a = 5;
    int b = 6;
    if(a > b)
    {
        Console.WriteLine("First number is greater.");
    }
else
    {
        Console.WriteLine("Second number is greater.)";
    }
}
```

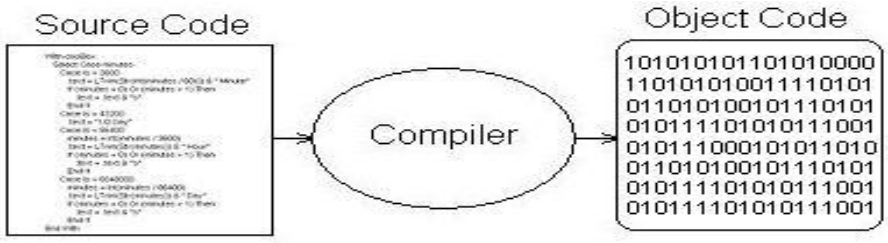
Source Code and Object Code

Source code

- The set of instructions written in any language other than machine language is called as source code.
- It is not directly understood by the machine (computer).

Object Code

- The set of instructions written in machine language is called as object code. It is also known as machine code.
- It is the only code which is directly understood by the machine (computer).



Language Translator

Language translator is a program that converts the source code in to the object code

Why Language Translator?

- Computer only understands object code (machine code).
- It does not understand any source code.
- There must be a program that converts source code in to the object code so that the computer can understand it.
- The language translator is one which does this job.
- The programmer writes the source code and then translator converts it in machine readable format (object code).

Types of Translators

Assembler

Compiler

Interpreter

Assembler – the language translator that converts assembly language code in to the object code (machine code).

Compiler – the language translator that converts high level language code in to the object code (machine code). It converts the whole code at a time.

Interpreter – the language translator that converts high level language code in to the object code (machine code). It converts the code line by line.