

COMPUTER LANGUAGES

- Programs should be written in Computer languages called programming Languages
- The programming languages consist of a set of characters & symbols
- There are specified rules of writing programs to communicate with computers.
- There are syntax rules to write the program which should be followed to obtain desired result.

Types of Programming languages

Machine language :

The Computer can understand only machine language which uses 0s & 1s. In machine language the different instructions to the computer are formed by taking different combinations of 0s & 1s.

This is lowest level of programming language in which instruction consists of two parts "opcode" & "operand"

| | |
|--------|---------|
| OPCODE | OPERAND |
|--------|---------|

The first part opcode (operation code) tells which operation is to be performed.

Second part operand tells the operation is to be performed on which data.

Advantages :

- ① Translation Free: Computer understands machine language so conversion is not required in case of machine language program.
- ② High Speed: As there is no need of translation Execution of machine language program is very fast.

Disadvantages :

- ① Complex Language: It is very hard to develop a program using machine language
- ② Error Prone: The chances of occurrence of error in machine level program is more as all coding is in the form of 0s & 1s.
- ③ Tedious: Writing program, making changes, corrections & modifications is complex task in Machine language.
- ④ Time Consuming: It is very time-consuming to write entire code in B. I. L.

Assembly language :

- To make programming easier, assembly language is developed which is logically equivalent to machine language, but is easy to read, write & understand.
- Assembly language is symbolic programming language that use symbolic notation to represent machine language instructions.
- These languages are known as low level language
- These languages uses mnemonic codes, which are easy to remember.

e.g. ADD for addition
 CMP for compare
 MUL for multiply
 STD for store

Syntax

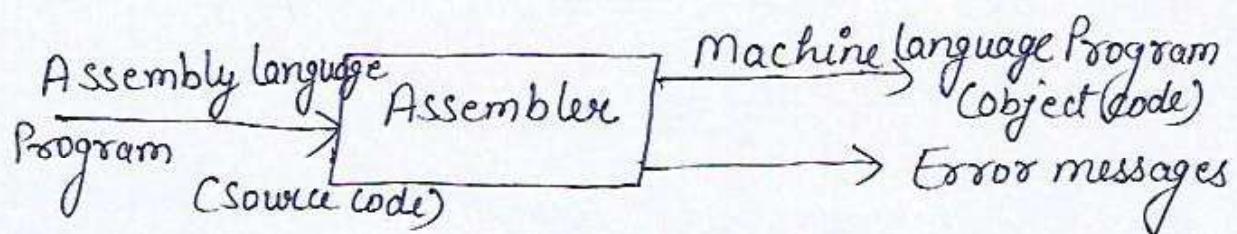
| label | Instruction (opcode) | Comment (operand) |
|-------|-------------------------|----------------------|
| START | mov | A, B |

copy B to A

- * label is symbolic name used to point instruction
- * Instruction specify opcode & operand
- * Comment is non executable part, specify the action.

Assembler

- Program written in assembly language require a translator to convert them into machine language.
- Assembler substitute the required machine code for each instruction in assembly language.



- The program written in assembly language is called source code & after converting it into machine language, it is called object code.
- Thus Assembler converts whole assembly language program into binary code & then it assemble the machine understandable code into main memory for execution.

Advantages of Assembly language:

- ① They are easy to understand and use.
- ② They are less error prone than machine language.
- ③ They require less resources than high level language hence more efficient.

Disadvantages

- ① Machine Dependent: Depend on architecture of computer
- ② Harder to learn: It is harder to learn as compared to high level languages
- ③ Development process is quite slow.
- ④ less efficient than machine language
- ⑤

High level Language :

A programming language in which the program statements are not closely related to internal characteristics of computer is called high-level language.

- These languages are machine independent.
- High level language fall somewhere close to general day to day language & makes programming easier & less error prone.
- e.g. C++, Java, Visual Basic etc.

Advantages :

- Programs are easily readable & understandable
- They are machine independent
- Faster & easier development
- Documentation is easy.

Disadvantages

- Poor control on hardware
- less efficient
- Porting program to new machine is not easy.