



Bharat Shikshan Sanstha's

**Shri Chhatrapati Shivaji College, Omerga**

Tq. Omerga Dist. Osmanabad - 413606 (MS), India

(Affiliated to Dr. Babasaheb Ambedkar Marathwada University, Aurangabad) | NAAC Reaccredited 'B' Grade

# Title of Paper : Computer Fundamental

Course Code : CS01

Class :B.Sc. First Year

Subject :Computer Sci.(Gen).

## Computer Languages

### Part-6

---

Dr. Revate S.S. (Asst. Professor & Head),  
Department of Computer Sci. & IT,

# Objective

---

- **What are programming languages and why need them.**
- **How computer languages are classifieds .**

## Computer Languages :

- Programs is the list of instructions to be executed by a computer are known as the software of a computer.
- 
- The electronic circuits used in building a computer that executes the software is known as the hardware of the computer.
  - Home Television set is hardware and various entertainment programs are software.
  - Hardware is a one-time expense whereas software is a continue expense.
  - Computer software may be classified into two categories : application software and system software.
  - Application software is the set of programs necessary to carry out operations for specified application.

## Computer Languages..

- For example , programs to solve a set of equations ,process of examinations results etc. are example of applications software.
- System software are general programs written for the system which provide the environment to facilitate the writing of application software.
- For example , operating system software, programming translators.
- A natural languages (Hindi, Marathi etc.) are medium of communication which is used for understanding ideas and sharing the information and also giving the instruction to each other.
- The computer programming languages used for giving instructions to the computer , so that it can do the required processing the task.

## Computer Languages..

- In computer science, **computer languages** are systems of communication with a computer. Such languages are used to create computer code or program code, the set of instructions forming a computer program which is executed by the computer.
- Programming languages are used in computer programming to implement algorithms.
- There are many programming languages which are used by computer professional for writing programs.
- Each programming language have its instruction set that allow to users to perform various specified task.
- The flexibility and the features of any programming language makes it different from other programming languages.

# Types of Programming Languages

- Programming languages are classified on the basis of their features ,their functions and capabilities.
- The classifications of programming languages as follows

---

  - Machine Languages(Low Level)
  - Assembly Languages
  - High Level Languages

## Machine Languages :

- Computers can be programmed to understand many different computer languages.
- Every computer understand only one language without using translation program known as machine language of computer.

---

- The machine language of a computer is written normally as strings of binary 1s and 0s.
- Machine code consists of a sequence of simple computer instructions with each instruction expressed as a string of binary digits or bits (i.e., 1's and 0's).
- For example one computer may represent the ADD operation as 10011111 while another might represent the same operations as 000110.
- This language is not easy to use because it is difficult to read and understand .

## Advantages of Machine Languages :

---

- Machine language program can be executed very fast by a computer.
- It requires no translator to translate the **code**.
- It is directly understood by the **computer**.



## Disadvantages of Machine Languages :

### 1) Machine dependent :

- As the internal design of every type computer is different from other type of computer ,so Machine language differs from computer to computer .
- Program written in machine language for one type of computer , cannot use to another type of computer, rewritten the program as per the computer type .

### 2) Difficult to program.

- It is necessary for a programmer to either memorize all machine's instruction set .
- A programmer is also to keep track of storage location of data and instructions.
- A machine language programmer must be knowledge about the hardware structure of the computer.

## Disadvantages of Machine Languages :

### 3) Error prone

- Writing programs in machine language , a programmer has to remember the opcodes and must keep track of storage locations of data and instructions.
- This makes it very difficult for him/her to concentrate fully on the logic of the problem, resulting in programming errors and removing the errors also difficult .

### 4) Difficult to modify.

- It is difficult to correct and modify the machine language programs.
- Checking machine instructions to locate errors is very difficult and time consuming.
- So modifying a machine language program is difficult.

## Assembly Languages :

- A language that allows instructions and storage location to be represented by letters and symbols is called assembly language or symbolic language.

---

- Examples : ADD,SUB,MUL,DIV etc.
- `ADD AX,BX ;Add values of BX into AX result in AX`
- A computer can directly execute only machine language program .Hence an assembly language program must be converted into its equivalent machine language program before it can be executed on the computer.
- This is done with help of a translator program called **assembler**.
- Assembler is the system software which translates an assembly language program into its equivalent machine program.
- It assembles program in main memory of the computer , and makes it ready for execution.

## Assembly Languages ..

- The process of translating an assembly language program into its equivalent machine program by using an assembler.
- Input to the assembler is the assembly language program ( a source program) , and its output is the machine language program ( as object program.).



Fig. A Process of Assembler

## Advantages of Assembly Languages :

1. Easier to understand : Due to the use of mnemonics instead of numeric op-code and symbolic name for data and locations instead of numeric address ,assembly language programs easier to understand .
2. Easier to locate and correct errors: Due to use of mnemonics op-codes and symbolic names for data locations ,programmers can easily locate errors and correct it.
3. Easier to modify : Since they are easier to understand ,it is easier to locate ,correct and modify instructions of an assembly language program.
4. No worry about address: It is not required to keep track of memory locations.
5. It is faster in speed, as its execution time is less.

## Disadvantages of Assembly Languages :

### 1) Machine dependent :

- As the internal design of every type computer is different from other type of computer ,so assembly language differs from computer to computer .
- Program written in assembly language for one type of computer , cannot use to another type of computer, rewritten the program as per the computer type .

### 2) Difficult to program.

- It is necessary for a programmer to either memorize all machine's instruction set .
- A machine language programmer must be knowledge about the hardware structure of the computer.

### 3) Machine level coding :

- One assembly language instructions is substituted for one machine language instructions . So it is also time consuming and difficult.

## High-level Language :

- High-level language uses English statements which are like human language, with less words also known as keywords and fewer ambiguities.
- Each high level language will have its own syntax and keywords.
- Machine and assembly are referred to as low-level programming languages .
- High-level programming languages were designed to overcome these limitations of low-level programming.
- A high-level language does not require addressing hardware constraints when developing a program.
- Every single program written in a high-level language must be converted into machine language before being executed by the computer. Because of computer understands only machine language.
- These conversion done by the language processors compiler or interpreter.
- BASIC, C/C++ and Java are popular examples of high-level languages.

## Advantages High-level Language :

1. The main **advantage of high-level languages** over **low-level languages** is that they are easier to read, write, and maintain.
2. High-level languages are machine independent. That is , a programmer written in a high-level can be ported and executed easily on any computer having a translator software for high level language.
3. Programmers do not require any internal structure of the computer to write the program in HLL.
4. They do not deal with machine-level coding . Rather ,they deal with High-level coding .So programmer writes the instructions in programs, using English words and mathematical symbols and expressions.
5. **Debugging is Easy** : High level language makes easier to write and debug a program and gives the programmer time to think about overall logic rather than thinking about program code. Because of the use of compilers and interpreters, errors can be detected and located easily.
6. Program development is faster

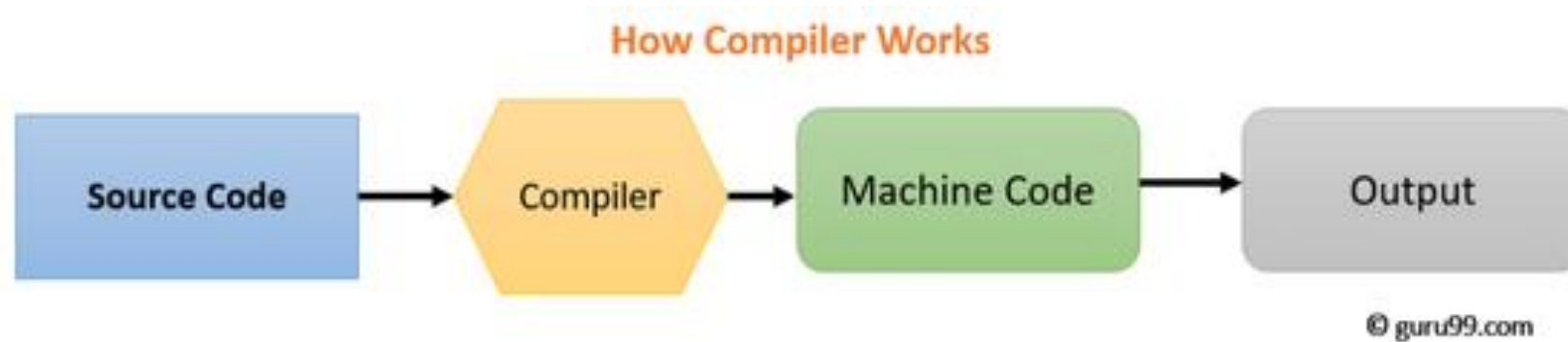


## Disadvantages High-level Language :

- A high level language program can't get executed directly.
- It requires some translator to get it translated to machine language. They are interpreter and compiler.
- **Slow Execution:** Program written in high level language need to be translated to machine language. This translation process increases the execution time of program.

# Compiler and Interpreter

- A high level language program can't get executed directly.
  - It requires some translator to get it translated to machine language.
  - They are interpreter and compiler.
- 



## Interpreter

- It is a special program and software which is used for translating a high level language program into machine language.
- The interpreter takes the high level language program code as input line by line and converts it into machine language , line by line ,without producing object code for future use.
- If the program needs to be rerun , then again it has to be interpreted and translated into the Machine language.
- The line by line is bit slow. It checks the single instructions for errors.(If error found ,translation and execution, stops .Otherwise it executes.)
- An interpreter make debugging a program easier because they can track errors line by line.
- BASIC, Perl, JavaScript etc. languages are used interpreter.

# Compiler

- It is that program , which converts the program in High level Language into machine language , and the object program is stored, which can be executed ,any time.
- These ~~compiler program~~ are efficient than the interpreter , as they take the whole source code and convert into reusable object code.
- Almost every high level language comes with own compilers .
- As compilers , convert total source code into object code, it is very fast to execute the program.
- The limitation of the use of compiler being , that whatever the source program is modified , it has to be again recompiled to include the changes into the object code.
- FORTRAN,COBOL ,C ,C++ etc. languages are used compiler.

# Interpreter Vs Compiler

## Interpreter

Translates program one statement at a time.

Interpreters usually take less amount of time to analyze the source code. However, the overall execution time is comparatively slower than compilers.

No intermediate object code is generated, hence are memory efficient.

Programming languages like JavaScript, Python, Ruby use interpreters.

## Compiler

Scans the entire program and translates it as a whole into machine code.

Compilers usually take a large amount of time to analyze the source code. However, the overall execution time is comparatively faster than interpreters.

Generates intermediate object code which further requires linking, hence requires more memory.

Programming languages like C, C++, Java use compilers.

## Loader

---

- Loader is the program which loads and object program into primary memory of the computer , and makes it ready for execution .
- The loader usually loads the programs ,relocates the links the object programs ,thereby making an object program executable.

# Linker

---

- Linker is the software ,which does the function of linking the object program to various modules , so that a single machine level program can be generated .
- This plays very important role as it links various library functions used ,various resources etc. , and helps the program to execute efficiently and effectively

Unit No.	Topic Name	Ref.	No. of Lecturers
1	<b>Fundamentals of Computer System :</b> Introduction, Characteristics & features of Computers, Components of Computers ,Organization of Computer.	1/1	3
2	<b>Algorithm and Flowcharts :</b> Algorithm: Definition , Characteristics, Advantages and disadvantages , Examples Flowchart : Definition ,Define symbols of flowchart ,Advantages and disadvantages ,Examples	2/1  3/3 3/4	3  3
3	<b>Computer Generation &amp; Classification</b> Generation of Computers : First to Fifth , Classification of Computers ,Distributed & Parallel computers	2/12	3
4	<b>Computer Languages</b> Types of Programming Languages :Machine Languages ,Assembly Languages ,High Level Languages • Assembler, Linker, Loader, Interpreter & Compiler	2/9  2/9	3



Unit No.	Topic Name	Ref.	No. of Lecturers
5	<b>Computer Memory :</b> <ul style="list-style-type: none"> <li>Memory Cell &amp; Organization</li> <li>Types of Memory (Primary And Secondary) : RAM ,ROM ,PROM ,EPROM, Secondary Storage Devices ( FD, CD, HD, Pen drive, DVD, Tape Drive, DAT )</li> </ul>	2/4 2/4	3
6	<b>I/O Devices:</b> Input Devices : Touch screen , OMR, OBR , OCR, Light pen , Scanners , Digitizers Output Devices :Plotters, LCD , Plasma Display, Printers	1/4 1/4	3
7	<b>Processor :</b> Structure of Instruction, Description of Processor , Processor Features ,RISC & CISC	2/5	6
8	<b>Operating system Concepts :</b> Why Operating System , Functions of Operating System, Types of Operating ,System , Batch O.S. , Multiprogramming O.S. , Time Sharing O.S ,Personal Computers O.S. ,Network O.S.	2/9 2/9	3
Core Ref.	1. Fundamentals of Information Technology By Chetan Srivastava, Kalyani Publishers 2. Fundamentals of Computers By V. Rajaraman, PHI Publication, IVth Edition. 3. Fundamentals of Programming By Raj K. Jain, S. Chand Publication		

## Fundamentals of Computer System

### Core References

1. Fundamentals of Information Technology By Chetan Srivastava, Kalyani Publishers
2. Fundamentals of Computers By V. Rajaraman, PHI Publication, IVth Edition.
3. Fundamentals of Programming By Raj K. Jain, S. Chand Publication

### Additional Reference:

1. Computer Today By Suresh K. Basandra, Galgotia Publication, Updated Edition
2. Computer Fundamental By B. Ram, BPB Publication.

*Thank you !*

---