

LANGUAGES, OPERATING SYSTEMS AND SOFTWARE PACKAGES

♣ Introduction

The computer language is defined as code or syntax which is used to write programs or any specific applications. The computer language is used to communicate with computers. Broadly the computer language can be classified into three categories assembly language, machine language, and high-level language. The machine language is considered as oldest computer language among all three. In machine language, the input is directly given as binary input which is processed by the machine. Binary inputs mean one and zero form.

For computer language processing the system needs compiler and interpreter to convert the language in computer language so that it can be processed by a machine.

Types of Languages

- o Machine Level Language**

- o Assembly Level Language**

- o High Level Language (3GL, 4GL, 5GL, etc.)**

1. Machine Language

The machine language is sometimes referred to as machine code or object code which is set of binary digits 0 and 1. These binary digits are understood and read by a computer system and interpret it easily. It is considered a native language as it can be directly understood by a central processing unit (CPU). The machine language is not so easy to understand, as the language uses the binary system in which the commands are written in 1 and 0 form which is not easy to interpret. There is only one language which is understood by computer language which is machine language. The operating system of the computer system is used to identify the exact machine language used for that particular system.

The operating system defines how the program should write so that it can be converted to machine language and the system takes appropriate action. The computer programs and scripts can also be written in other programming languages like C, C++, and JAVA. However, these languages cannot be directly understood by a computer system so there is a need for a program that can convert these computer programs to machine language. The compiler is used to convert the programs to machine language which can be easily understood by computer systems. The compiler generates the binary file and executable file.

Example of machine language for the text “Hello World”.

```
01001000 0110101 01101100 01101100 01101111 00100000 01010111 01101111  
01110010 01101100 01100100.
```

2. Assembly Language

The assembly language is considered a low-level language for microprocessors and many other programmable devices. The assembly language is also considered as second-generation language. The first generation language is machine language. The assembly language is mostly famous for writing an operating system and also in writing different desktop applications. The operations carried out by programmers using assembly language are memory management, registry access, and clock cycle operations. The drawback of assembly language is the code cannot be reused and the language is not so easy to

understand. The assembly language is considered a group of other languages. It is used to implement the symbolic representation of machine code which is used to program CPU architecture. The other name of assembly language is assembly code. For any processor, the most used programming language is assembly language.

In assembly language, the programmer does the operation which can be directly executed on a central processing unit (CPU). The language has certain drawbacks as it does not contain any variables or functions in programs and also the program is not portable on different processors. The assembly language uses the same structure and commands which machine language does use but it uses names in place of numbers. The operations performed using the assembly language is very fast. The operations are much faster when it is compared to high-level language.

3. High-Level Language

The development of high-level language was done when the programmers face the issue in writing programs as the older language has portability issues which mean the code written in one machine cannot be transferred to other machines. Thus lead to the development of high-level language. The high-level language is easy to understand and the code can be written easily as the programs written are user-friendly in a high-level language. The other advantage of code written in a high-level language is the code is independent of a computer system which means the code can be transferred to other machines. The high-level of language uses the concept of abstraction and also focus on programming language rather than focusing on computer hardware components like register utilization or memory utilization.

The development of higher-level language is done for a programmer to write a human-readable program that can be easily understood by any user. The syntax used and the programming style can be easily understood by humans if it is compared to low-level language. The only requirement in a high-level language is the need of compiler. As the program written in a high-level language is not directly understood by the computer system. Before the execution of high-level programs, it needs to be converted to machine level language. The examples of high-level language are C++, C, JAVA, FORTRAN, Pascal, Perl, Ruby, and Visual Basic.

♣ Translator (Assembler / Compiler / Interpreter)

Language Processors –

Assembly language is machine dependent yet mnemonics that are being used to represent instructions in it are not directly understandable by machine and high Level language is machine independent. A computer understands instructions in machine code, i.e. in the form of 0s and 1s. It is a tedious task to write a computer program directly in machine

code. The programs are written mostly in high level languages like Java, C++, Python etc. and are called **source code**. These source code cannot be executed directly by the computer and must be converted into machine language to be executed. Hence, a special translator system software is used to translate the program written in high-level language into machine code is called **Language Processor** and the program after translated into machine code (object program / object code).

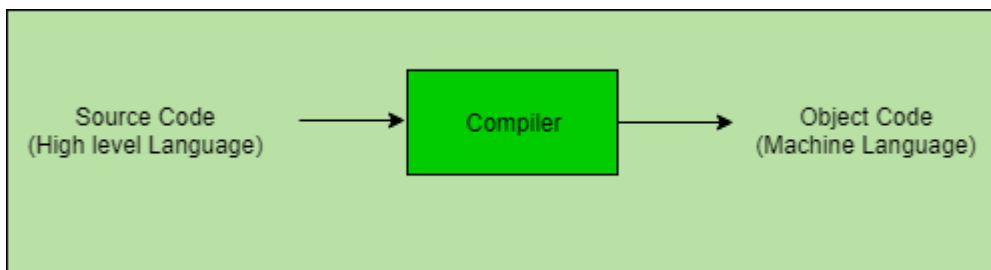
The language processors can be any of the following three types:

Compiler

The language processor that reads the complete source program written in high level language as a whole in one go and translates it into an equivalent program in machine language is called as a Compiler.

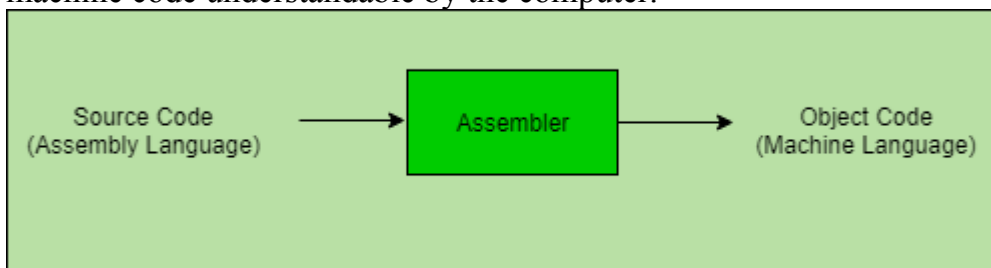
Example: C, C++, C#, Java

In a compiler, the source code is translated to object code successfully if it is free of errors. The compiler specifies the errors at the end of compilation with line numbers when there are any errors in the source code. The errors must be removed before the compiler can successfully recompile the source code again.>



Assembler–

The Assembler is used to translate the program written in Assembly language into machine code. The source program is a input of assembler that contains assembly language instructions. The output generated by assembler is the object code or machine code understandable by the computer.



Interpreter–

The translation of single statement of source program into machine code is done by language processor and executes it immediately before moving on to the next line is called an interpreter. If there is an error in the statement, the interpreter terminates its translating process at that statement and displays an error message. The interpreter moves on to the next line for execution only after removal of the error. An Interpreter

directly executes instructions written in a programming or scripting language without previously converting them to an object code or machine code.
Example: Perl, Python and Matlab.

Difference between Compiler and Interpreter –

Compiler	Interpreter
A compiler is a program which converts the entire source code of a programming language into executable machine code for a CPU.	interpreter takes a source program and runs it line by line, translating each line as it comes to it.
Compiler takes large amount of time to analyze the entire source code but the overall execution time of the program is comparatively faster.	Interpreter takes less amount of time to analyze the source code but the overall execution time of the program is slower.
Compiler generates the error message only after scanning the whole program, so debugging is comparatively hard as the error can be present any where in the program.	Its Debugging is easier as it continues translating the program until the error is met
Generates intermediate object code.	No intermediate object code is generated.
Examples: C, C++, Java	Examples:-- Perl, Python , Ruby

♣ Types of Operating Systems

- o Batch Operating System**
- o Multi Processing Operating System**
- o Time Sharing Operating System**
- o Online and Real Time Operating System**

Operating systems are there from the very first computer generation and they keep evolving with time. In this chapter, we will discuss some of the important types of operating systems which are most commonly used.

Batch operating system

The users of a batch operating system do not interact with the computer directly. Each user prepares his job on an off-line device like punch cards and submits it to the computer operator. To speed up processing, jobs with similar needs are batched together and run as a group. The programmers leave their programs with the operator and the operator then sorts the programs with similar requirements into batches.

The problems with Batch Systems are as follows –

- Lack of interaction between the user and the job.

- CPU is often idle, because the speed of the mechanical I/O devices is slower than the CPU.
- Difficult to provide the desired priority.

Time-sharing operating systems

Time-sharing is a technique which enables many people, located at various terminals, to use a particular computer system at the same time. Time-sharing or multitasking is a logical extension of multiprogramming. Processor's time which is shared among multiple users simultaneously is termed as time-sharing.

The main difference between Multi-programmed Batch Systems and Time-Sharing Systems is that in case of Multi-programmed batch systems, the objective is to maximize processor use, whereas in Time-Sharing Systems, the objective is to minimize response time.

Multiple jobs are executed by the CPU by switching between them, but the switches occur so frequently. Thus, the user can receive an immediate response. For example, in a transaction processing, the processor executes each user program in a short burst or quantum of computation. That is, if n users are present, then each user can get a time quantum. When the user submits the command, the response time is in few seconds at most.

The operating system uses CPU scheduling and multiprogramming to provide each user with a small portion of a time. Computer systems that were designed primarily as batch systems have been modified to time-sharing systems.

Advantages of Timesharing operating systems are as follows –

- Provides the advantage of quick response.
- Avoids duplication of software.
- Reduces CPU idle time.

Disadvantages of Time-sharing operating systems are as follows –

- Problem of reliability.
- Question of security and integrity of user programs and data.
- Problem of data communication.

Distributed operating System

Distributed systems use multiple central processors to serve multiple real-time applications and multiple users. Data processing jobs are distributed among the processors accordingly.

The processors communicate with one another through various communication lines (such as high-speed buses or telephone lines). These are referred as loosely coupled

systems or distributed systems. Processors in a distributed system may vary in size and function. These processors are referred as sites, nodes, computers, and so on.

The advantages of distributed systems are as follows –

- With resource sharing facility, a user at one site may be able to use the resources available at another.
- Speedup the exchange of data with one another via electronic mail.
- If one site fails in a distributed system, the remaining sites can potentially continue operating.
- Better service to the customers.
- Reduction of the load on the host computer.
- Reduction of delays in data processing.

Network operating System

A Network Operating System runs on a server and provides the server the capability to manage data, users, groups, security, applications, and other networking functions. The primary purpose of the network operating system is to allow shared file and printer access among multiple computers in a network, typically a local area network (LAN), a private network or to other networks.

Examples of network operating systems include Microsoft Windows Server 2003, Microsoft Windows Server 2008, UNIX, Linux, Mac OS X, Novell NetWare, and BSD.

The advantages of network operating systems are as follows –

- Centralized servers are highly stable.
- Security is server managed.
- Upgrades to new technologies and hardware can be easily integrated into the system.
- Remote access to servers is possible from different locations and types of systems.

The disadvantages of network operating systems are as follows –

- High cost of buying and running a server.
- Dependency on a central location for most operations.
- Regular maintenance and updates are required.

Real Time operating System

A real-time system is defined as a data processing system in which the time interval required to process and respond to inputs is so small that it controls the environment. The time taken by the system to respond to an input and display of required updated

information is termed as the response time. So in this method, the response time is very less as compared to online processing.

Real-time systems are used when there are rigid time requirements on the operation of a processor or the flow of data and real-time systems can be used as a control device in a dedicated application. A real-time operating system must have well-defined, fixed time constraints, otherwise the system will fail. For example, Scientific experiments, medical imaging systems, industrial control systems, weapon systems, robots, air traffic control systems, etc.

There are two types of real-time operating systems.

Hard real-time systems

Hard real-time systems guarantee that critical tasks complete on time. In hard real-time systems, secondary storage is limited or missing and the data is stored in ROM. In these systems, virtual memory is almost never found.

Soft real-time systems

Soft real-time systems are less restrictive. A critical real-time task gets priority over other tasks and retains the priority until it completes. Soft real-time systems have limited utility than hard real-time systems. For example, multimedia, virtual reality, Advanced Scientific Projects like undersea exploration and planetary rovers, etc.

♣ Uses and applications of Software Packages

- o Word Processing Packages
- o Spread Sheet Packages
- o Graphical Packages
- o Database Packages I
- o Presentation Packages
- o Animation / Video / Sound Packages

Definition

Software is the means by which computer systems speak with computer users. Software forms the heart of computer systems. What are the major types of software? Read on to find out.

Major Types of Software

(1) Programming Software:

This is one of the most commonly known and popularly used forms of computer software. These software come in forms of tools that assist a programmer in writing computer programs. Computer programs are sets of logical instructions that make a computer system perform certain tasks. The tools that help the programmers in instructing a computer system include text editors, compilers and interpreters.

(2) System Software:

It helps in running the computer hardware and the computer system. System software is a collection of operating systems; device drivers, servers, windowing systems and utilities. System software helps an application programmer in abstracting away from hardware, memory and other internal complexities of a computer.

(3) Application Software :

It enables the end users to accomplish certain specific tasks. Business software, databases and educational software are some forms of application software. Different word processors, which are dedicated for specialized tasks to be performed by the user, are other examples of application software.

Apart from these three basic types of software, there are some other well-known forms of computer software like inventory management software, ERP, utility software, accounting software and others. Take a look at some of them.

(4) Inventory Management Software:

This type of software helps an organization in tracking its goods and materials on the basis of quality as well as quantity. Warehouse inventory management functions encompass the internal warehouse movements and storage. Inventory software helps a company in organizing inventory and optimizing the flow of goods in the organization, thus leading to an improved customer service.

(5) Utility Software:

Also known as service routine, utility software helps in the management of computer hardware and application software. It performs a small range of tasks. Disk defragmenters, systems utilities and virus scanners are some of the typical examples of utility software.

(6) Data Backup and Recovery Software:

An ideal data backup and recovery software provides functionalities beyond simple copying of data files. This software often supports user needs of specifying what is to be backed up and when. Backup and recovery software preserve the original organization of files and allow an easy retrieval of the backed up data.

This was an overview of the major types of software. Computer software are widely popular today and hence we cannot imagine a world of computers without them. We would not have been able to use computers if not for the software. What is fascinating about the

world of computers is that it has its own languages, its ways of communication with our human world and human interaction with the computers is possible, thanks to computer software. I wonder, if the word 'soft' in 'software' implies 'soft-spokenness', which is an important quality.

1- Application Software

A set of programs used to solve particular problems of user through computer is called Application software. It is also known as application package. The ready packages are also available in market on CDs for various purposes. These are used by user who does not know the computer programming. The users solve their problems by using ready packages more easily and quickly. Some of application packages are: 1. Word processing software. 2. Spreadsheet software. 3. Database Management System software. 4. Graphics software. 5. Communication software.

1. Word Processing Software

This software is used to create and to edit documents such as letters, reports, essays etc. The word processing software provides several features for document editing and formatting. In editing process, text is entered into the document, deleted, copied or moved to another location etc. In formatting process, different formats can be applied on the text to make the document more attractive before to print on the printer. The most popular word processing software programs are: Microsoft Word, Word Perfect etc.

2. Spreadsheet Software

Spreadsheet software is used to store and process data in an electronic sheet having columns and rows. The data is entered into the cells of the sheet. The intersection of a row and a column is known as cell. Each cell is a unique address. The numbers and formulas are entered into the cells and the computer can automatically perform the calculation on numerical data in cells.

The spreadsheet software is commonly used for business application such as for performing financial calculations and recording transactions. The most popular example of spreadsheet software is Microsoft Excel etc.

3. Database Management Software

Database management software is used to create and manage databases. A database is a collection of related information or records on any subject such as records of the books in a library, information about the students of a college etc.

Database Management Software stores and manages records in databases. These records can be accessed very quickly when required. The Microsoft Access, Oracle etc. are most popular examples of Database Management Software.

4. Presentation Graphics Software

The presentation graphic software is used to create slides for making presentations. The presentation graphic software also has pre-drawn clip art images, which can be inserted into slides and can be modified. The PowerPoint is an example of presentation graphic software.

5. Communication Software

The Communication software is used to exchange information electronically. It is most commonly used software to send and receive information on the Internet. It also allows for sending and receiving faxes directly. The computer files can also be transferred from one PC to another through this software. The Internet Explorer is an example of this software.