IT 231
# Foundation of Information Technology

Unit 4

Computer Software

Software

→Instructions that tell a computer what to perform.

→Set of instruction or code use to execute task in computers.

→Computer Software is a collection of data or computer programs that tell the computer how to work.

- A computer system consists of hardware and software. The computer hardware cannot perform any task on its own. It need to be instructed about the tasks to be performed. Software is a set of computer programs together with necessary, documentation, configuration files and setup files that instructs the computer about the tasks to be performed.

- Software is defined as a collection of programs, documentation and operating procedures. The Institute of Electrical and Electronic Engineers (IEEE) defines software as a 'collection of computer programs, procedures, rules and associated documentation and data.

- Software controls, integrates, and manages the hardware components of a computer system. It also instructs the computer what needs to be done to perform a specific task and how it is to be done. For example, software instructs the hardware how to print a document, take input from the user, and display the output.

Classification of Software
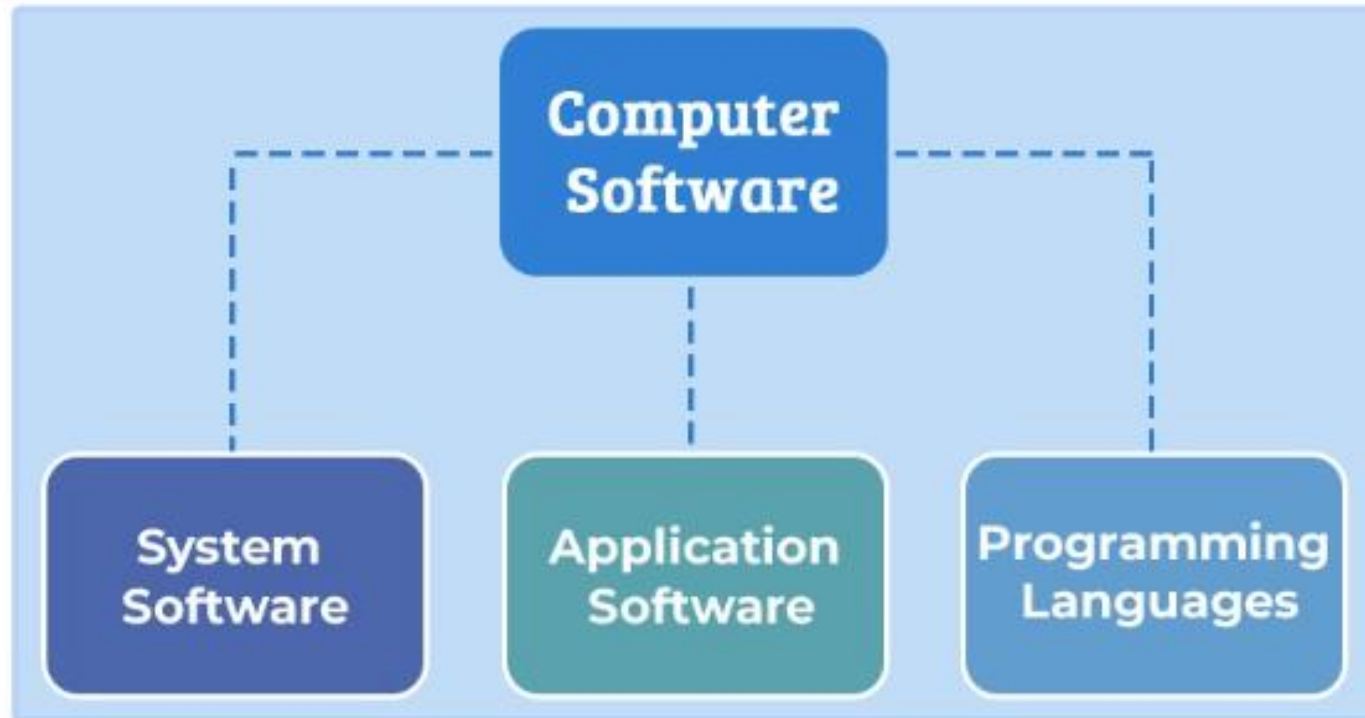
System Software

- It is a group of programs that direct the internal operations of computer system such as controlling I/O devices, managing the storage area within the computer, etc

- The purpose of the system software is to make the use of computer more efficient and easier.

- It allows the users to communicate with hardware system.

- It includes operating systems, and utilities for managing computer resources

Type of System Software

- Operating System

- Utility Software

- Device Driver

Operating System

- It is a program that controls the execution of application programs and acts as an interface between the user of the computer and computer hardware

- It is an integrated set of programs that is used to manage the various resources and overall operation of the computer system.

- It is the program running at all time on the computer.

- The primary objective of O/S are:

  - Making computer easy to use

  - Managing the resources of computer system

Cont..

- Operating System is a software, which makes a computer to actually work

- It is the software the enables all the programs we use.

- It boots the computer, launches application software, and ensures that all actions requested by a user are valid and processed in an orderly fashion.

- Examples: Windows, Linux, Unix and Mac OS, etc.,

OS Structure

Application Programs

System Programs

Software (Operating System)

HARDWARE
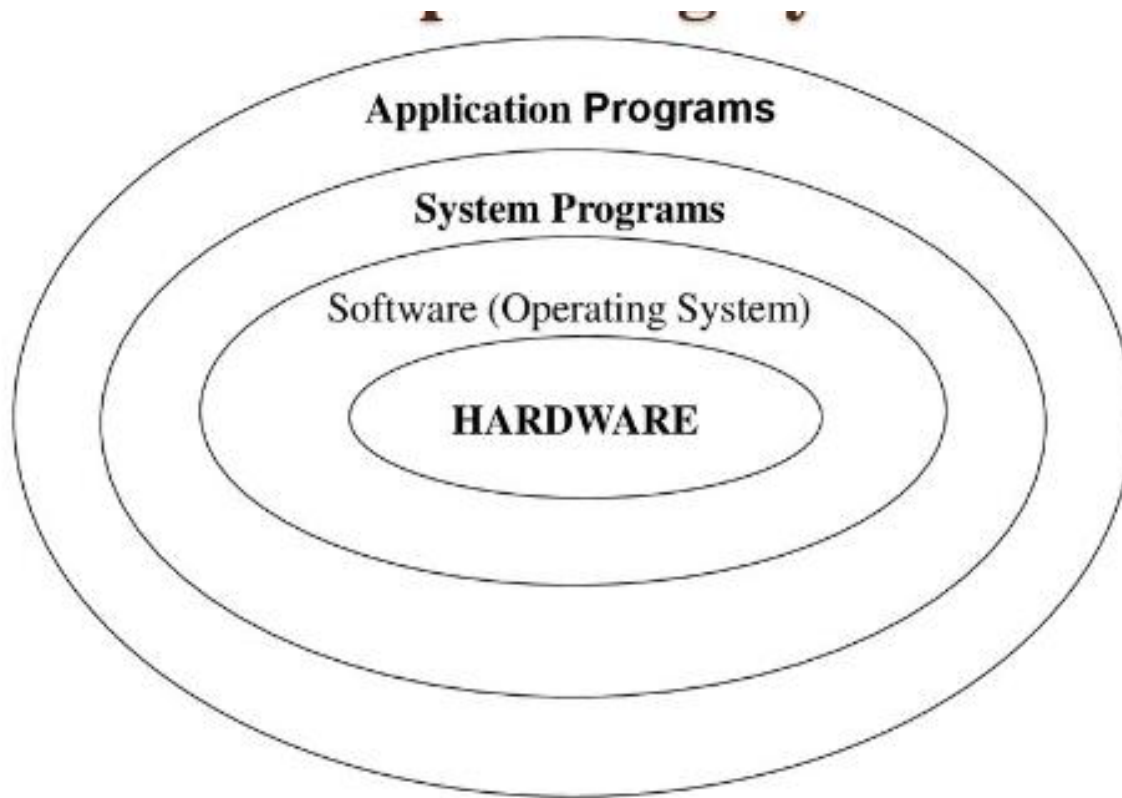
Function of Operating System

- Process Management
- Memory Management
- File Management
- Device Management
- Protection and Security
- User Interface or Command Interpreter

Process Management

▶ When a program stored in secondary storage device is loaded into memory, it will be ready for execution and is termed as process.

▶ Simply, we can say that a program in action is called process.

▶ All these processes compete for CPU and some other shared resources, and also need to communicate with each other.

▶ OS does the following activities for process management.
  ▶ Process Scheduling
  ▶ Keeps status of processes
  ▶ Interprocess communication

Memory Management

▶ It refers to management of Primary Memory or Main Memory which provides a fast storage that can be accessed directly by the CPU.

▶ For a program to be executed, it must be in the main memory.

▶ OS does the following activities for memory management.

   ▶ Managing free space

   ▶ Allocating and deallocating memory for processes

   ▶ Managing virtual memory

13

File Management

- A file system is normally organized into directories for easy navigation and usage.

- These directories may contain files and other directories.

- OS does the following activities for file management.

  - Naming files

  - Defining file structure

  - Defining file access methods

  - Defining file attributes and operations

  - Storing and handling files

14

Device Management

▶ Hardware devices typically provide the ability to input data into the computer or output data from the computer.

▶ OS does the following activities for device management.

  ▶ Handling input/output devices

  ▶ Monitoring the status of each device such as printers etc.

  ▶ Handling errors

Security and Protection

- OS is responsible for providing proper authentication and authorization to computer system and its resources.

- Authentication is the process of verifying identity of users by means of passwords before allowing to use computer system and its resources.

- Authorization is the process of providing access rights or privileges for objects to users which is implemented by using access control list or capability lists.

Command Interpretation

- As a user communicates with computer system using a specified set of commands, any command given by the user is first interpreted by the OS then appropriate action is taken

Types of OS

- The various types of OS areas follows:
1. Batch processing
2. Single user
3. Multi user
4. Multi tasking
5. Multi programming
6. Multi processing
7. Time sharing
8. Real Time

Batch Processing

- It is also known as off line or sequential or stacked processing system.
- In this OS, jobs with similar needs are batched together by the operator and they are stored on card and run as a group on a computer system.
- Users did not interact directly with the computer systems, but he prepared a job (comprising of the program, the data, & some control information) and submit it to operator. When output produced is submitted to the appropriate user.
- It is appropriate for executing large job that need little interaction.

Single User OS

- These OS allow only one user to operate at a time i.e. DOS.

- These OS are mainly used on personal computers. In DOS we cannot run another program at same time. For this we will have to close the first program, them only would be able to work on another program.

- Windows 95, 98 are single user OS but they support multi- tasking and multi-programming.

Multi User OS

- A multi-user operating system is a computer operating system (OS) that allows multiple users on different computers or terminals to access a single system with one OS on it.

- These programs are often quite complicated and must be able to properly manage the necessary tasks required by the different users connected to it. The users will typically be at terminals or computers that give them access to the system through a network, as well as other machines on the system such as printers.

- A multi-user operating system differs from a single-user system on a network in that each user is accessing the same OS at different machines. Example: Linux, Unix

Multi Tasking OS

- Multitasking is a logical extension of multiprogramming system that supports multiple programs to run concurrently. In multitasking more than one task are executed at the same time. In this technique the multiple tasks, also known as processes, share common processing resources such as a CPU. You may write a report with one program while another program plays a music.

- To do this work the OS directs the processor to spend a predetermined amount of time executing the instruction for each program at a time. Thus a small part of the first program is being processed and then the processor moves to the remaining programs one at a time , processing small part of each program and it is repeated until the processing is completed

Multi Programming OS

► Multiprogramming is a technique to execute number of programs simultaneously by a single processor.

► In   Multiprogramming, number of processes reside in main memory at a time and hence memory is also utilized.

► The CPU picks and begins to executes one of the jobs in the main memory. If   any I/O wait happened in a process,   then   CPU switches from that job to another job.

► Hence CPU in not idle at any time.

► In this way a single CPU is in demand.

Multi Processing OS

▶ Multiprocessor Operating System refers to the use of two or more central processing units (CPU) within a single computer system. These multiple CPUs are in a close communication sharing the computer bus, memory and other peripheral devices.

▶ Two possible approaches for multiprocessing are co-processing and parallel processing. In co-processing each of which handles a particular task. In parallel processing several full-fledged processors work together on the same tasks, sharing memory.

Time sharing OS

- A time sharing system allows many users to share the computer resources simultaneously. In other words, time sharing refers to the allocation of computer resources in time slots to several programs simultaneously.
- The time sharing systems were developed to provide an interactive use of the computer system. A time shared system uses CPU scheduling and multiprogramming to provide each user with a small portion of a time-shared computer. It allows many users to share the computer resources simultaneously.

Time sharing OS

- The OS allocates a set of time to each user. When this time is expired, it passes control to the next user on the system. The time allowed is extremely small and the users are given the impression that they each have their own CPU and they are the sole owner of the CPU.
- There are many tasks and activities that should be run at the same time. Like in smartphones the notification bar, games, songs, weather forecasting are all running at the same time so time sharing systems are more efficient to use

Real time OS

▶ The main objective of real time OS is their quick and predictable response to events. Response time is major parameter for producing the result.

▶ A real time OS is considered to function correctly only if it returns the correct result within any time constraint.

▶ Processing must be done within fixed time constraint

▶ If output must be produced on exact duration then it is said to be hard real time OS which is used in air traffic control system. If output can be produced on approximate duration and some [27]deadline can be missed then it is said to be soft real time OS.

Utility Software

- It is a computer program that performs a very specific task, usually related to managing system resources. It is also called system support software or service program which provides useful services to the user of the computer.

- It assists in maintaining, managing and protecting computer system resources.

- Utilities are used to merge and sort sets of data, keep track of computer jobs being run and perform other important routine tasks.

- Utility programs often come installed on computer systems; a number of utility programs can also be purchased.

Utility Software

- Disk Defragmentation and disk utilities: maintain files on disk and arrange them in contiguous manner for fast access.
- Backup utilities: safeguard files by creating backup copies
- Security Software: search for and remove viruses and spyware from computers and guard against attacks and it must be update frequently.
- System restore: restore computer to a previous state, if a problem occurs without loosing personal data files.
- Spam and pop-up blockers: save users time by eliminating unwanted junk mail and ads.

Utility Software

- Compression utilities: output a shorter stream or smaller file by compressing them.
- Network managers: checks network, log events and check data transfer
- Windows Cleaners: remove unwanted programs and leftover traces of program from the system.
- Parental controls: filter internet content and place restrictions on computer use for safety.

Utility Software

- For managing and transferring files
- File management utilities: provide tools for copying, deleting, renaming and organizing files.
- CD/DVD burners: copy and store files on CDs and DVDs
- File transfer utilities: move and share file across networks
- Search utilities: find files
  - Major utility software are:
- Device Driver
- Antivirus

Device Driver

▶ The OS often provide complete programs for working with special drivers such as printer, scanner, DVD ROM, etc. These programs are known as Device Driver as they allow the OS and other programs to activate and use.

▶ A device driver or a software driver is a computer program that allows higher-level computer programs to interact with a hardware device.

▶ A driver communicate with the device through the computer bus or communicates subsystem to which the hardware is connected.

▶ Programs which wants to use the device invokes a routine in the driver and the driver issues commands to the device. Once the device sends data back to the driver, the driver may invoke routine in the original calling program.

Device Driver

- Drivers are hardware dependent and OS specific.

- The main purpose of device driver is to simplify programming by acting as a translator between a device and the applications or OS that use it.

- Drivers are used for interfacing with printers, image scanners, digital cameras and many more.

33

Application Software

▶ It is computer program that performs a specific function directly for the users. It helps the user to work faster, more efficiently and more productively.

▶ They are designed to run on an OS for fulfilling various purposes and functionalities.

▶ They are developed and supplied by software companies as per the requirement.

▶ It can be categorized into two types:

    ▶ Customized or Tailored Software

    ▶ Packaged Software

Customized or Tailored SOftware

- ▶ It is the application software which is designed to fulfill the specific requirements of an organization.

- ▶ It is developed on the demand of a customer by a software contractor. Example: school billing system, hospital management system, result processing software, etc.

Packaged Software

- It is generalized set of programs designed and developed for the general purpose.

- Example: Microsoft Office, Adobe package, accounting package, etc

New Trends in software

- ▶ Low-code & no-code software development: Low-code and no-code options can be utilized to complement software development pipelines by allowing users to develop, deploy, and manage some parts of software solutions and delivery pipelines.

- ▶ Big data security: Big data and data science have become the norm in the IT industry, with data the cornerstone of any business. Software development has evolved to cater to big data needs from collecting, storing, and analyzing data.

- ▶ Increasing reliance on artificial intelligence: AI has already become a core component in most software, from simple computer vision applications to enterprise-scale predictive analytics.

New Trends in software

- Growth in IoT: The Internet of Things is ever-expanding, with billions of smart devices powering many industries across the world, ranging from simple home appliances to medical devices. Thus, more and more software will be developed targeting these IoT devices.

- Augmented reality (AR): This can be the most widely used technology as it has applications across most industries, from e-Commerce to changing the way users shop to navigation with AR- powered GPS apps.

Programming Languages

▶ A program is a set of sequenced instructions to cause a computer to perform particular operations or to solve a given problem. Thus, a computer requires a program to operate.

▶ The process of developing such program is called programming.

▶ The language that is used to develop the program is called programming language.

▶ A person who does programming is called programmer.

▶ A programming language or computer language is a standardized communication technique for giving instructions to a computer. It is set of syntax and semantic rules used to define computer programs.

Programming Languages

▶ There are many programming language among which a programmer may select any one. But this program must be translated to the computer understandable form.

▶ So, a language translator used to translate a computer program written by a programmer into a machine understandable form.

▶ Some of the programming languages are:

▶ 1. Machine Language or First Generation Language (1GL)

▶ 2. Assembly Language or Second Generation Language 2GL)

▶ 3. Procedural Language or Third Generation Language (3GL)

▶ 4. Problem Oriented Language or Fourth Generation Language (4GL)

▶ 5. Natural Language or Fifth Generation Language (5GL)

Machine Language

- A programming language in which every statement of the program is written in terms of binary 0s and 1s, that is directly understood by the computer without any translation is called machine language.

- The entire circuitry of a computer is designed in such a way that it understands the machine language instructions and converts them into electrical signal required to run the computer.

- They are the first generation language. An instruction written in any machine language has two parts:

- Operation Code (Opcode)

- Operands

Machine Language

▶ An opcode is a command word for an operation such as add, compare, or jump. The operand for an instruction specifies the data or the address of the data for the operation.

▶ Example: if 01100001 is a computer instruction for ADD, then to add 2 and 3, we have to write 01100001 (i.e. ADD) 00000010 (i.e.2) 00000011 (i.e. 3)

Assembly Language

- When symbols are employed for the operation part and address parts of the instruction code, the representation is called an assembly language program.

- This is second generation language (2GL).

- Machine and assembly languages are referred to as low level language since the coding for a problem is at the individual instruction level.

- Each machine has got its own assembly language, which is dependent upon the internal architecture of the processor.

- A machine cannot execute program written in an assembly level language, as it is not in binary form

Assembly Language

▶ An assembler is a translator, which takes its input in the form of an assembly language program and produces machine code as output.

▶ The assembly language program is referred as source program and the machine language program is known as object program

Assembly Language Program → Assembler → Object Code in Machine Language
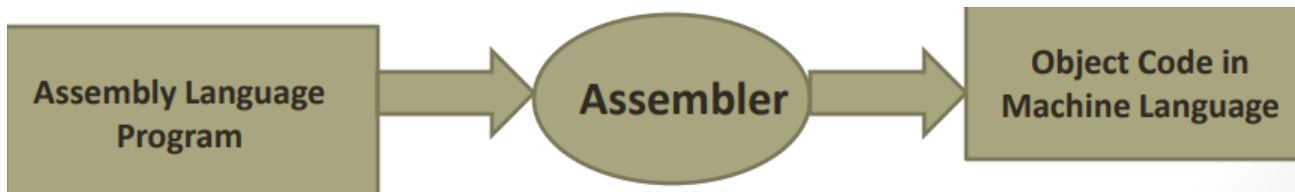
Low level Languages

- ▶ It is the computer programming language that is close to the machine instruction.

- ▶ Low level languages are simple, but are considered difficult to use due to difficulty in remembering the numerous technical details.

- ▶ These are machine dependent programming languages such as Binary (Machine code) and Assembly language.

- ▶ Since computer only understand the Binary language that means instructions in the form of 0"s and 1"s (Signals - that can be either High or Low), so these programming languages are the best way to give signals (Binary Instructions) to the computer directly. - Machine Code (Binary Language) does not need a interpreter or compiler to convert language in any form because computer understands these signals directly

Low level Languages

▶ But, Assembly language needs to be converted in equivalent Binary code, so that computer can understand the instructions written in Assembly. Assembler is used to convert an assembly code to its equivalent Binary code. - The codes written in such kind of languages are difficult to write, read, edit and understand; the programs are not portable to any other computer system.

▶ Low Level programming language programs are faster than High Level programming language programs as they have less keywords, symbols and no need (less need) to convert into Machine Code

High Level Languages

- ▶ These languages are called higher level languages because their syntax is closer to human language than it is to assembly or machine language code.

- ▶ These are generally problem oriented and machine independent. It does not use mnemonic codes.

- ▶ These are using a set of word and symbols following some rules similar to natural language such as English.

- ▶ The programs written in high level languages are known as source code and these codes are converted into machine code by using interpreter or compiler.

- ▶ To express computer operations, these languages use operators such as plus or minus sign, that are familiar to mathematics.

High Level Languages

▶ Therefore, reading, writing and understanding computer programs is easier with a high level language, although the instructions must still be translated into machine language.

▶ The languages like Java, .Net, Pascal, COBOL, C++, C, C# and other (which are very popular now to develop user end applications). These languages come under the high level programming language category.

▶ High level programming languages have some special keywords, functions and class libraries by using them we can easily build a program for the computer

Types of HIGH level Languages

- Procedure Oriented Language(3GL)
  - COBOL, FORTRAN, C, C++, Java, etc
- Problem Oriented Language(4GL)
  - SQL, Visual Basic
- Natural Language(5GL)
  - PROLOG, etc

Assignment

- Write short notes on :
    - Translator
    - Assemblers
    - Compilers
    - Interpreter