Introduction of Operating System

An operating system acts as an intermediary between the user of a computer and computer hardware. The purpose of an operating system is to provide an environment in which a user can execute programs in a convenient and efficient manner.

An operating system is software that manages the computer hardware. The hardware must provide appropriate mechanisms to ensure the correct operation of the computer system and to prevent user programs from interfering with the proper operation of the system.

Operating System – Definition:

- An operating system is a program that controls the execution of application programs and acts as an interface between the user of a computer and the computer hardware.
- A more common definition is that the operating system is the one program running at all times on the computer (usually called the kernel), with all else being application programs.

An operating system is concerned with the allocation of resources and services, such as memory, processors, devices, and information. The operating system correspondingly includes programs to manage these resources, such as a traffic controller, a scheduler, memory management module, I/O programs, and a file system.

Operating system as User Interface –

1. User
2. System and application programs
3. Operating system
4. Hardware

Every general-purpose computer consists of the hardware, operating system, system programs, and application programs. The hardware consists of memory, CPU, ALU, and I/O devices, peripheral device, and storage device. System program consists of compilers, loaders, editors, OS, etc. The application program consists of business programs, database programs.

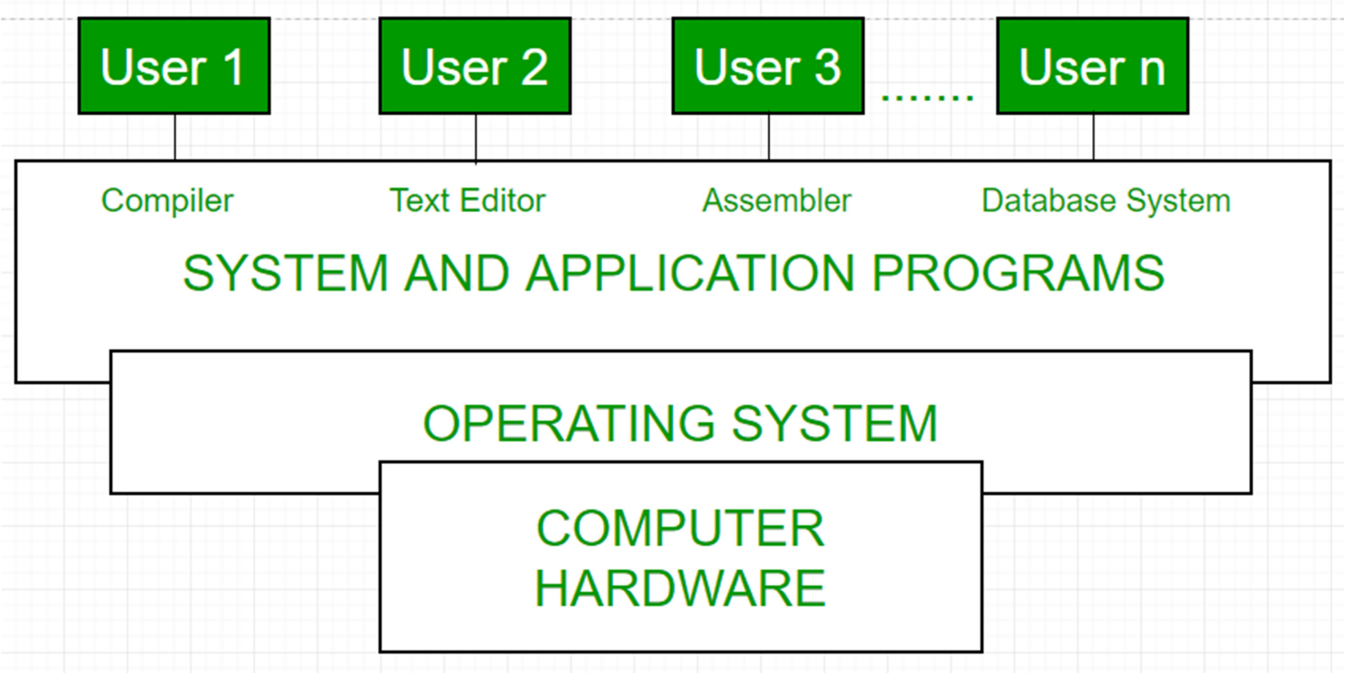


Fig1: Conceptual view of a computer system

Every computer must have an operating system to run other programs. The operating system coordinates the use of the hardware among the various system programs and application programs for various users. It simply provides an environment within which other programs can do useful work.

The operating system is a set of special programs that run on a computer system that allows it to work properly. It performs basic tasks such as recognizing input from the keyboard, keeping track of files and directories on the disk, sending output to the display screen and controlling peripheral devices. OS is designed to serve two basic purposes:

1. It controls the allocation and use of the computing System's resources among the various user and tasks.
2. It provides an interface between the computer hardware and the programmer that simplifies and makes feasible for coding, creation, debugging of application programs.

Generations of Operating System

The First Generation (1940 to early 1950s)

When the first electronic computer was developed in 1940, it was created without any operating system. In early times, users have full access to the computer machine and write a program for each task in absolute machine language. The programmer can perform and solve only simple mathematical calculations during the computer generation, and this calculation does not require an operating system.

The Second Generation (1955 - 1965)

The first operating system (OS) was created in the early 1950s and was known as **GMOS. General Motors** has developed OS for the **IBM** computer. The second-generation operating system was based on a single stream batch processing system because it collects all similar jobs in groups or batches and then submits the jobs to the operating system using a punch card to complete all jobs in a machine. At each completion of jobs (either normally or abnormally), control transfer to the operating system that is cleaned after completing one job and then continues to read and initiates the next job in a punch card. After that, new machines were called mainframes, which were very big and used by professional operators.

The Third Generation (1965 - 1980)

During the late 1960s, operating system designers were very capable of developing a new operating system that could simultaneously perform multiple tasks in a single computer program called multiprogramming. The introduction of **multiprogramming** plays a very important role in developing operating systems that allow a CPU to be busy every time by performing different tasks on a computer at the same time. During the third generation, there was a new development of minicomputer's phenomenal growth starting in 1961 with the DEC PDP-1. These PDP's leads to the creation of personal computers in the fourth generation.

The Fourth Generation (1980 - Present Day)

The fourth generation of operating systems is related to the development of the personal computer. However, the personal computer is very similar to the minicomputers that were developed in the third generation. The cost of a personal computer was very high at that time; there were small fractions of minicomputers costs. A major factor related to creating personal computers was the birth of Microsoft and the Windows operating system. Microsoft created the first **window** operating system in 1975. After introducing the Microsoft Windows OS, Bill Gates and Paul Allen had the vision to take personal computers to the next level. Therefore, they introduced the **MS-DOS** in 1981; however, it was very difficult for the person to understand its cryptic commands. Today, Windows has become the most popular and most commonly used operating system technology. And then, Windows released various operating systems such as Windows 95, Windows 98, Windows XP and the latest operating system, Windows 7. Currently, most Windows users use the Windows 10 operating system. Besides the Windows operating system, Apple is another popular operating system built in the 1980s, and this operating system was developed by Steve Jobs, a co-founder of Apple. They named the operating system Macintosh OS or Mac OS.

Batch Processing :

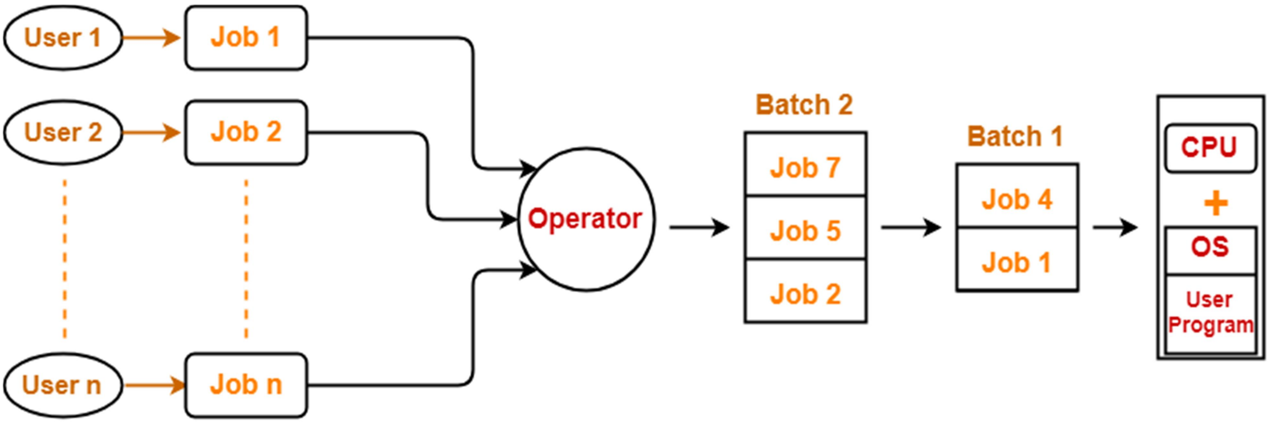
A series of jobs are executed without any human intervention in Batch processing system. In this set of jobs with similar needs are batched together and inputted to the computer for execution. It is also called as Simple Batch System. It is slower in processing than Multiprogramming system.

Advantages of Batch Processing :

- It manages large repeated work easily.
- No special hardware and system support required to input data in batch systems.
- It can be shared by multiple users.
- Very less idle time of the batch system.
- Enables us to manage the efficiently large load of work.

Disadvantages of Batch Processing :

- It has more turnaround time.
- Non linear behavior.
- Irreversible behavior.
- Due to any mistake, it may happen any job can go infinite loop.
- proves to be costly sometimes.



Batch Operating System

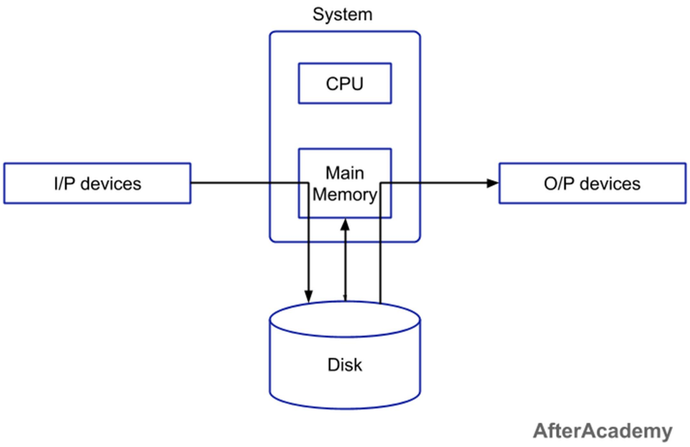
SPOOLING:-

SPOOL is an acronym for **simultaneous peripheral operations on-line**. It is a kind of buffering mechanism or a process in which data is temporarily held to be used and executed by a device, program or the system. Data is sent to and stored in memory or other volatile storage until the program or computer requests it for execution.

In a computer system peripheral equipment's, such as printers and punch card readers etc (batch processing), are very slow relative to the performance of the rest of the system. Getting input and output from the system was quickly seen to be a bottleneck. Here comes the need for spool.

Spooling works like a typical request queue where data, instructions and processes from multiple sources are accumulated for execution later on. Generally, it is maintained on computer's physical memory, buffers or the I/O device-specific interrupts. The spool is processed in FIFO manner i.e. whatever first instruction is there in the queue will be popped and executed.

Spooling stands for "**Simultaneous Peripheral Operations Online**". So, in a Spooling, more than one I/O operations can be performed simultaneously i.e. at the time when the CPU is executing some process then more than one I/O operations can also de done at the same time. The following image will help us in understanding the concept in a better way:



From the above image, we can see that the input data is stored in some kind of secondary device and this data is then fetched by the main memory. The benefit of this approach is that, in general, the CPU works on the data stored in the main memory. Since we can have a number of input devices at a time, so all these input devices can put the data into the disk or secondary memory. Then, the main memory will fetch the data one by one from the secondary memory and the CPU will execute some instruction on that data. Both the main memory and secondary memory are digital in nature, so taking data from the main to secondary is very fast. Also, when the CPU is executing some task then at that time, the input devices need not wait for its turn. They can directly put their data in the secondary memory without waiting for its turn. By doing so, the CPU will be in the execution phase most of the time. So, the CPU will not be idle in this case.

When the CPU generates some output, then that output is first stored in the main memory and the main memory transfers that output to the secondary memory and from the secondary memory, the output will be provided to some output devices. By doing so, again we are saving time because now the CPU doesn't have to wait for the output device to show the output and this, in turn, increases the overall execution speed of the system. The CPU will not be held idle in this case.

For example, in a printer spooling, there can be more than one documents that need to be printed. So, the documents can be stored into the spool and the printer can fetch that documents and print the document one by one.

Advantages of Spooling

- Since there is no interaction of I/O devices with CPU, so the CPU need not wait for the I/O operation to take place. The I/O operations take a large amount of time.
- The CPU is kept busy most of the time and hence it is not in the idle state which is good to have a situation.

- More than one I/O devices can work simultaneously.

Difference between Spooling and Buffering

We all know that a buffer is an area in the main memory that is used to store and hold data temporarily. This data can be transferred between two devices or between a device and an application. The main aim of buffers is to match the speed of data streaming between a sender and receiver. There is a difference between Spooling and Buffering.

- In spooling, the I/O of one job can be handled along with some operations of another job. While in buffering, only one job is handled at a time.
- Spooling is more efficient than buffering.
- In buffering, there is a small separate area in the memory know as a buffer. But spooling can make use of the whole memory.

Multiprogramming :

Multiprogramming operating system allows to execute multiple processes by monitoring their process states and switching in between processes. It executes multiple programs to avoid CPU and memory underutilization. It is also called as Multi-program Task System. It is faster in processing than Batch Processing system.

Advantages of Multiprogramming :

- CPU never becomes idle
- Efficient resources utilization
- Response time is shorter
- Short time jobs completed faster than long time jobs
- Increased Throughput

Disadvantages of Multiprogramming :

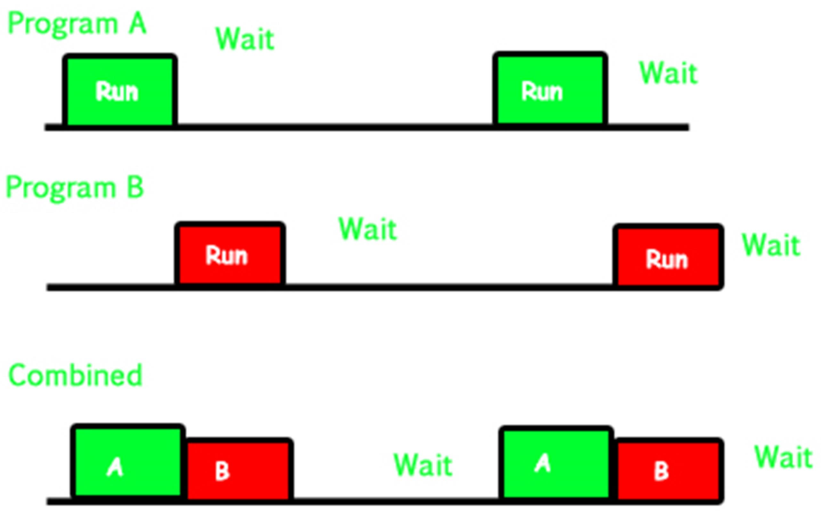
- Long time jobs have to wait long
- Tracking all processes sometimes difficult
- CPU scheduling is required
- Requires efficient memory management
- User interaction not possible during program execution

The main idea of multi programming is to maximize the CPU time.

Multi programmed system's working –

- In a multi-programmed system, as soon as one job goes for an I/O task, the Operating System interrupts that job, chooses another job from the job pool (waiting queue), gives CPU to this new job and starts its execution. The previous job keeps doing its I/O operation while this new job does CPU bound tasks. Now say the second job also goes for an I/O task, the CPU chooses a third job and starts executing it. As soon as a job completes its I/O operation and comes back for CPU tasks, the CPU is allocated to it.
- In this way, no CPU time is wasted by the system waiting for the I/O task to be completed. Therefore, the ultimate goal of multi programming is to keep the CPU busy as long as there are processes ready to execute. This way, multiple programs can be executed on a single processor by executing a part of a program at one time, a part of another program after this, then a part of another program and so on, hence executing multiple programs. Hence, the CPU never remains idle.

In the image below, program A runs for some time and then goes to waiting state. In the mean time program B begins its execution. So the CPU does not waste its resources and gives program B an opportunity to run.



Multiprocessing –

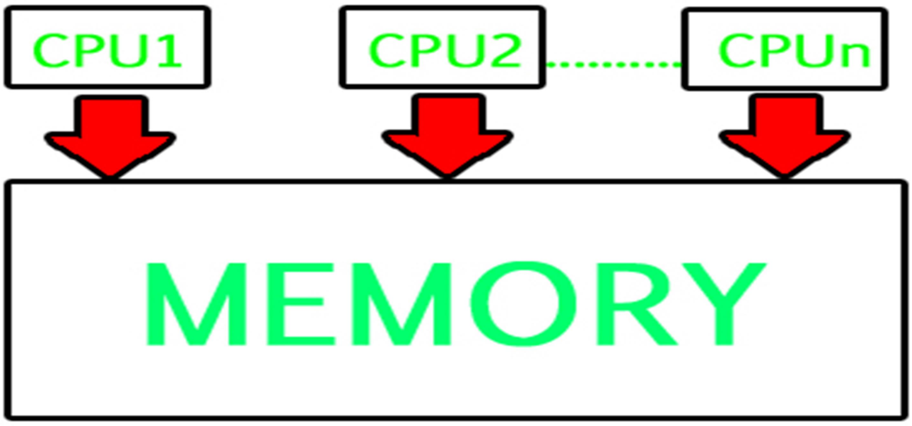
In a uni-processor system, only one process executes at a time. Multiprocessing is the use of two or more CPUs (processors) within a single Computer system. The term also refers to the ability of a system to support more than one processor within a single computer system. Now since there are multiple processors available, multiple processes can be executed at a time. These multi processors share the computer bus, sometimes the clock, memory and peripheral devices also.

Multi processing system's working –

- With the help of multiprocessing, many processes can be executed simultaneously. Say processes P1, P2, P3 and P4 are waiting for execution. Now in a single processor system, firstly one process will execute, then the other, then the other and so on.
- But with multiprocessing, each process can be assigned to a different processor for its execution. If its a dual-core processor (2 processors), two processes can be executed simultaneously and thus will be two times faster, similarly a quad core processor will be four times as fast as a single processor.

Why use multi processing –

- The main advantage of multiprocessor system is to get more work done in a shorter period of time. These types of systems are used when very high speed is required to process a large volume of data. Multi processing systems can save money in comparison to single processor systems because the processors can share peripherals and power supplies.
- It also provides increased reliability in the sense that if one processor fails, the work does not halt, it only slows down. e.g. if we have 10 processors and 1 fails, then the work does not halt, rather the remaining 9 processors can share the work of the 10th processor. Thus the whole system runs only 10 percent slower, rather than failing altogether.



Multiprocessing refers to the hardware (i.e., the CPU units) rather than the software (i.e., running processes). If the underlying hardware provides more than one processor then that is multiprocessing. It is the ability of the system to leverage multiple processors' computing power.

Difference between Multi programming and Multi processing –

- A System can be both multi programmed by having multiple programs running at the same time and multiprocessing by having more than one physical processor. The difference between multiprocessing and multi programming is that Multiprocessing is basically executing multiple processes at the same time on multiple processors, whereas multi programming is keeping several programs in main memory and executing them concurrently using a single CPU only.
- Multiprocessing occurs by means of parallel processing whereas Multi programming occurs by switching from one process to other (phenomenon called as context switching).

Multitasking OR Time Sharing –

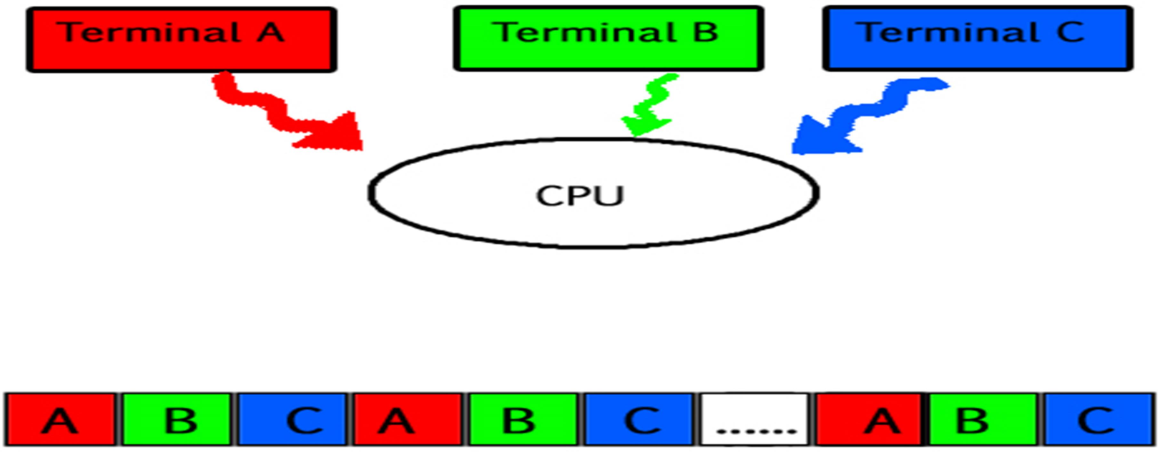
As the name itself suggests, multi tasking refers to execution of multiple tasks (say processes, programs, threads etc.) at a time. In the modern operating systems, we are able to play MP3 music, edit documents in Microsoft Word, surf the Google Chrome all simultaneously, this is accomplished by means of multi tasking.

Multitasking is a logical extension of multi programming. The major way in which multitasking differs from multi programming is that multi programming works solely on the concept of context switching whereas multitasking is based on time sharing alongside the concept of context switching.

Multi tasking system's working –

- In a time sharing system, each process is assigned some specific quantum of time for which a process is meant to execute. Say there are 4 processes P1, P2, P3, P4 ready to execute. So each of them are assigned some time quantum for which they will execute e.g time quantum of 5 nanoseconds (5 ns). As one process begins execution (say P2), it executes for that quantum of time (5 ns). After 5 ns the CPU starts the execution of the other process (say P3) for the specified quantum of time.
- Thus the CPU makes the processes to share time slices between them and execute accordingly. As soon as time quantum of one process expires, another process begins its execution.
- Here also basically a context switch is occurring but it is occurring so fast that the user is able to interact with each program separately while it is running. This way, the user is given the illusion that multiple processes/ tasks are executing simultaneously. But actually only one process/ task is executing at a particular instant of time. In multitasking, time sharing is best manifested because each running process takes only a fair quantum of the CPU time.

In a more general sense, multitasking refers to having multiple programs, processes, tasks, threads running at the same time. This term is used in modern operating systems when multiple tasks share a common processing resource (e.g., CPU and Memory).



- As depicted in the above image, At any time the CPU is executing only one task while other tasks are waiting for their turn. The illusion of parallelism is achieved when the CPU is reassigned to another task. i.e all the three tasks A, B and C are appearing to occur simultaneously because of time sharing.
- So for multitasking to take place, firstly there should be multiprogramming i.e. presence of multiple programs ready for execution. And secondly the concept of time sharing.

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.

Functions of an Operating System

To achieve the goals of an Operating system, the Operating System performs a number of functionalities. They are:

*** Process Management:** At a particular instant of time, the CPU may have a number of processes that are in the ready state. But at a time, only one process can be processed by a processor. So, the CPU should apply some kind of algorithm that can be used to provide uniform and efficient access to resources by the processes. The CPU should not give priority to only one process and it should make sure that every process which is in the ready state will be executed. Some of the CPU scheduling algorithms are First Come First Serve, Round Robin, Shortest Job First, Priority Scheduling, etc.

*** Memory Management:** For the execution of a process, the whole process is put into the main memory and the process is executed and after the execution of the process, the memory is freed and that memory can be used for other processes. So, it is the duty of the Operating System to manage the memory by allocating and deallocating the memory for the process.

*** I/O Device Management:** There are various I/O devices that are present in a system. Various processes require access to these resources and the process should not directly access these devices. So, it is the duty of the Operating System to allow the use of I/O devices by the various process that are requiring these resources.

*** File Management:** There are various files, folders and directory system in a particular computer. All these are maintained and managed by the Operating System of the computer. All these files related information are maintained by using a File Allocation Table or FAT. So, every detail related to the file i.e. filename, file size, file type, etc is stored in the File Allocation Table. Also, it is the duty of the Operating System to make sure that the files should not be opened by some unauthorized access.

*** Virtual Memory:** When the size of the program is larger than the main memory then it is the duty of the Operating System to load only frequently used pages in the main memory. This is called Virtual Memory.

Goals of the Operating System

There are two types of goals of an Operating System i.e. Primary Goals and Secondary Goal.

*** Primary Goal-User convenience:** The primary goal of an Operating System is to provide a user-friendly and convenient environment. We know that it is not compulsory to use the Operating System, but things become harder when the user has to perform all the process scheduling and converting the user code into machine code is also very difficult. So, we make the use of an Operating System to act as an intermediate between us and the hardware. All you need to do is give commands to the Operating System and the Operating System will do the rest for you. So, the Operating System should be convenient to use.

*** Secondary Goal-Efficient use of a Computer System:** The secondary goal of an Operating System is efficiency. The Operating System should perform all the management of resources in such a way that the resources are fully utilized and no resource should be held idle if some request to that resource is there at that instant of time.

Operating System as resource manager:- A computer system has many resources (hardware and software), which may be require to complete a task. The commonly required resources are input/output devices, memory, file storage space, CPU etc. The operating system acts as a manager of the above resources and allocates them to specific programs and users, whenever necessary to perform a particular task. Therefore operating system is the resource manager i.e. it can manage the resource of a computer system internally. The resources are processor, memory, files, and I/O devices. In simple terms, an operating system is the interface between the user and the machine.

Operating system is like a government; In a next manner, the operating system is like a government , the government collects the money from various services (Public, companies, Taxes etc) and distribute the money to different development activities.

Same as the OS collects the resources from the network environment as to a system, and grant the resources to requesting jobs.

A computer has many resources (Hardware and Software), which may be required to complete a task. The commonly required resources are Input/Output devices, Memory file storage space, CPU(Central Processing Unit) time and so on.

The operating system acts as the manager of these resources and allocates them to specific programs and users as necessary for their tasks. Therefore we can say an operating system is a resource allocator. This is the main features of an operating system.

When a number of computers connected through a network more than one computers trying for a computer print or a common resource, then the operating system follow same order and manage the resources in an efficient manner.

Generally, resources sharing in two ways "in time" and "in space". When a resource is a time-sharing resource, first one of the tasks get the resource for some time, then another and so on.

For example, a CPU in the time-sharing system. In the time-sharing system, OS fixes the time slot for the CPU. First one of the processes gets the CPU, when the time slot expires, the CPU switch to the next process in the ready queue. In this example, the CPU is a time resource. CPU Scheduling in Operating System

The other kind of sharing is the "space sharing". In this method, the users share the space of resource. For example, the main memory consisting of several processes at a time, so the process of sharing the resources.

The main difference between "in time" sharing resources and "in space" sharing resources is "in time" resources are not divided into units, whereas "in space" sharing resources are divided into units.

Operating System as an Abstract machine:-

Abstract Machine:- An abstract machine is a model of a computer system (considered either as hardware or software) constructed to allow a detailed and precise analysis of how the computer system works. Such a model usually consists of input, output, and operations that can be preformed (the operation set), and so can be thought of as a processor. Turing machines are the best known abstract machines, but there exist many other machines as well such as cellular automata.

Abstract machines that model software are usually thought of as having very high-level operations. For example, an abstract machine that models a banking system can have operations like "deposit," "withdraw," "transfer," etc.

An abstract machine implemented in software is termed a virtual machine, and one implemented in hardware is called simply a "machine."

Extends the basic hardware with added functionality Provides high-level abstractions:

- More programmer friendly
- Common core for all applications
  - – E.g. File system instead of just registers on a disk controller

It hides the details of the hardware

- Makes application code portable