**An operating system** acts as an intermediary between the user of a computer and the 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. Internally, operating systems vary greatly in their makeup since they are organized along many different lines of code. The operating system (OS) is in-charge of making sure the system operates correctly and efficiently in an easy-to-use manner. These goals form the basis for choices among various algorithms and strategies. Because an operating system is large and complex, it must be created piece by piece, such as virtualization, concurrency, and persistence. What an OS does: it takes physical resources, such as a CPU, memory, or disk, and virtualizes them. It handles tough and tricky issues related to concurrency. And it stores files persistently, thus making them safe over the long-term. Each of these pieces should be a well-delineated portion of the system, with carefully defined inputs, outputs, and functions.

**An operating system** is a program that manages a computer’s hardware. It also provides a basis for application programs and acts as an intermediary between the computer user and the computer hardware. An amazing aspect of operating systems is how they vary in accomplishing these tasks. Mainframe operating systems are designed primarily to optimize utilization of hardware. Personal computer (PC) operating systems support complex games, business applications, and everything in between. Operating systems for mobile computers provide an environment in which a user can easily interface with the computer to execute programs. Thus, some operating systems are designed to be convenient, others to be efficient, and others to be some combination of the two. Before we can explore the details of computer system operation, we need to know something about system structure. The study of Operating System concentrates on the basic functions of system startup, I/O, and storage. Also focuses on the basic computer architecture that makes it possible to write a functional operating system.

Virtualization is a general technique that adopted by Operating System, where it takes a physical resource (such as the processor, or memory, or a disk) and transforms it into a more general, powerful, and easy-to-use virtual form of itself. Thus, we sometimes refer to the operating system as a virtual machine.

The **Operating System** makes use of the features of the virtual machine (such as running a program, or allocating memory, or accessing a file), the OS also provides some interfaces (APIs) that you can call. A typical OS, in fact, exports a few hundred system calls that are available to applications. Because the OS provides these calls to run programs, access memory and devices, and other related actions. Virtualization allows many programs to run (thus sharing the CPU), and many programs to concurrently access their own instructions and data (thus sharing memory), and many programs to access devices (thus sharing disks and so forth), the OS is sometimes known as a resource manager. Each of the CPU, memory, and disk is a resource of the system; it is thus the operating system’s role is to manage those resources, doing so efficiently or fairly or indeed with many other possible goals in mind.

An **Operating Systems** make systems relatively easy to use, and virtually all operating systems you use today have been influenced by many developments, where the final goal is to provide an environment in which we can execute programs, to make the computer system convenient to use, and to make the use of computer hardware in efficient way.

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 conveniently and efficiently.

An **operating system** is a software that manages 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.

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 always running 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, a memory management module, I/O programs, and a file system.

**Functions of Operating System**

1. **Memory Management:** One of the main functions of OS is to manage the primary and secondary memory. All the memory devices such as hard disk, pen drive etc. are managed by OS. Memory management monitors each memory location, in any case either it is allocated, or it is not allocated (free). Memory allocation to the processes is also decided and checked by Operating System. It decides and checks which process will obtain memory and at what time. The operating system manages the Primary Memory or Main Memory. Main memory is made up of a large array of bytes or words where each byte or word is assigned a certain address. Main memory is fast storage, and it can be accessed directly by the CPU. For a program to be executed, it should be first loaded in the main memory. It keeps track of primary memory, i.e., which bytes of memory are used by which user program. The memory addresses that have already been allocated and the memory addresses of the memory that has not yet been used. In multiprogramming, the OS decides the order in which processes are granted access to memory, and for how long. It Allocates the memory to a process when the process requests it and deallocates the memory when the process has terminated or is performing an I/O operation.
2. **Device Management:** The operating system manages the communication between the computer system and the peripheral devices connected to the I/O port of the system. Peripheral devices use their respective drivers to communicate with the system. The operating system determines which program or process is accessed by which connection and device. It also makes sure when a program is executed or terminated, it will stop the communication between the device and the computer system. An OS manages device communication via their respective drivers. It keeps track of all devices connected to the system. designates a program responsible for every device known as the Input/Output controller. Decides which process gets access to a certain device and for how long. Allocates devices in an effective and efficient way. Deallocates devices when they are no longer required.
3. **Process Management:** In a multi-programming environment, the OS decides the order in which processes have access to the processor, and how much processing time each process has. This function of OS is called process scheduling. An Operating System performs the following activities for processor management. It keeps track of the status of processes. The program which performs this task is known as a traffic controller. Allocates the CPU that is a processor to a process. De-allocates processor when a process is no more required. The process is a program under the execution. The operating system manages all the processes so that each process gets the CPU for a specific time to execute itself, and there will be less waiting time for each process.
4. **Security:** Computer security is a very important aspect of any operating system. The reliability of an operating system is determined by how much better security it provides us. Modern operating systems use a firewall for security. A firewall is a security system that monitors every activity happening in the computer and blocks that activity in case of any threat. Access **Authorization** and Authentication are the two major components for security in operating system. Authorization is the process of giving someone permission to do or have something. **Authentication** mechanism determines the user’s identity before revealing the sensitive information. It is very crucial for the system or interfaces where the operating system priority is to protect the confidential information.

Authentication refers to identifying each user of the system and associating the executing programs with those users. It is the responsibility of the Operating System to create a protection system which ensures that a user who is running a particular program is authentic. Operating Systems generally identifies/authenticates users three ways that is **Username / Password** where the User need to enter a registered username and password with Operating system to login into the system. **User card/key** where User need to punch card in card slot, or enter key generated by key generator in option provided by operating system to login into the system. **User attribute - fingerprint/ eye retina pattern/ signature** where user need to pass his/her attribute via designated input device used by operating system to login into the system.

1. **Error Detection:** The Operating System is responsible for the detection of any type of error or bugs that can occur while any task. The operating system constantly monitors the system to detect errors and avoid the malfunctioning of a computer system.  The well-secured OS sometimes also acts as a countermeasure for preventing any sort of breach to the Computer System from any external source and probably handling them. While a computer system is running, a variety of errors might occur. Error detection guarantees that data is delivered reliably across susceptible networks. Errors may occur in the CPU and memory hardware (such as a memory error or a power failure), in I/O devices (such as a parity error on disk, a connection failure on a network, or lack of paper in the printer), and in the user program (such as an arithmetic overflow, an attempt to access an illegal memory location, or a too-great use of CPU time). For each type of error, the operating system should take the appropriate action to ensure correct and consistent computing. Sometimes, it has no choice but to halt the system. At other times, it might terminate an error-causing process or return an error code to a process for the process to detect and possibly correct. The operating system continuously monitors the system to locate or recognize problems and protects the system from them.
2. **Co-ordination between Software and Users:** The operating system (OS) allows hardware components to be coordinated and directs and allocates assemblers, interpreters, compilers, and other software to different users of the computer system.
3. **Job Accounting:** Operating system Keeps track of all the functions of a computer system, it also tracks of time and resources used by various tasks and users, this information can be used to track resource usage for a particular user or group of users. Hence, it makes a record of all the activities taking place on the system. It has an account of all the information about the memory, resources, errors, etc. Therefore, this information can be used as and when required.
4. **File Management:** A file system is organized into directories for efficient or easy navigation and usage. These directories may contain other directories and other files. An Operating System carries out the many file management activities such as, It keeps track of where information is stored, user access settings and status of every file, and more… These facilities are collectively known as the file system. A file is collection of specific information stored in the memory of computer system. File management is defined as the process of manipulating files in computer system, its management includes the process of creating, modifying, and deleting the files. The file management of function in operating system (OS) is based on the concepts such as File Attributes, File Operations, File Access Permissions and File Systems.

File Attributes specifies the characteristics of the files such as type, date of last modification, size, location on disk etc. file attributes help the user to understand the value and location of files. File attributes is one most important feature. It is uses to describe all the information regarding file. File Operations specifies the task that can be performed on a file such as opening and closing of file. File Access permission specifies the access permissions related to a file such as read write and Execute.

File Systems specifies the logical method of file storage in a computer system. Some of the commonly used file systems include FAT and NTFS.

**Types of Operating System**

1. **Batch Operating System:** In this system, the OS does not forward the jobs/tasks directly to the CPU. It works by grouping together similar types of jobs under one category. Further, we name this group as a ‘batch’. Hence, the name batch OS. Batch processing was very popular in the 1970s. The jobs were executed in batches. People used to have a single computer known as a mainframe. Users using batch operating systems do not interact directly with the computer. Each user prepares their job using an offline device like a punch card and submitting it to the computer operator. Jobs with similar requirements are grouped and executed as a group to speed up processing. Once the programmers have left their programs with the operator, they sort the programs with similar needs into batches. The batch operating system grouped jobs that perform similar functions. These job groups are treated as a batch and executed simultaneously. A job is a single unit that consists of a pre-set sequence of commands, data, and programs. Processing takes place in the order in which they are received, i.e., first come, first serve. These jobs are stored in memory and executed without the need for manual information. When a job is successfully run, the operating system releases its memory.

**Batch Operating System is of majorly two types they are**:

1. **Single Batch Operating System**

The user did not directly interact with the computer system for job execution in a simple batch operating system. However, the user was required to prepare a job that included the program, control information, and data on the nature of the job on control cards. The job was then submitted to the computer operator, who was usually in the form of a punch card. The program's output included results and registers and memory dumps in the event of a program error. The output appeared after some time that could take days, hours, and minutes.

Its main role was to transfer control from one job to another. Jobs with similar requirements were pooled together and processed through the processor to improve processing speed. The operators were used in the program to create batches with similar needs. The computer runs the batches one by one when they became available. This system typically reads a sequence of jobs, each with its control cads and predefined job tasks.

1. **multi-programmed batched System**

Spooling deals with many jobs that have already been read and are waiting to run on disk. A disk containing a pool of jobs allows the operating system to choose which job to run next to maximize CPU utilization. Jobs that come on magnetic tape or cards directly cannot be run in a different order. Jobs run sequentially because they are executed in a first-come, first-served manner. When various jobs are stored on a direct access device, job scheduling becomes possible like a disk. Multi-programming is an important feature of job scheduling. For overlapped I/O, spooling and offline operations have their limitations. Generally, a single user could not maintain all the input/output devices, and CPU buys at all times.

In the multi-programmed batched system, jobs are grouped so that the CPU only executes one job at a time to improve CPU utilization. The operating system maintains various jobs in memory at a time. The operating system selects one job and begins executing it in memory. Finally, the job must wait for a task to complete, such as mounting a tape on an I/O operation. In a multiprogramming system, do not sit idle because the operating system switches to another task. When a job is in the wait state, and the current job is completed, the CPU is returned.

1. **Multi – Programming System**

A multiprogramming operating system may run many programs on a single processor computer. If one program must wait for an input/output transfer in a multiprogramming operating system, the other programs are ready to use the CPU. As a result, various jobs may share CPU time. However, the execution of their jobs is not defined to be at the same time-period. The primary goal of multiprogramming is to manage the entire system's resources. The key components of a multiprogramming system are the file system, command processor, transient area, and I/O control system. As a result, multiprogramming operating systems are designed to store different programs based on sub-segmenting parts of the transient area. The resource management routines are linked with the operating system core functions.

The Multi-Programming Systems are of two types are i, Multitasking Operating System

ii. Multiuser Operating System

A multitasking [operating system](https://www.javatpoint.com/os-tutorial) enables the execution of two or more programs at the same time. The operating system accomplishes this by shifting each program into and out of memory one at a time. When a program is switched out of memory, it is temporarily saved on disk until it is required again.

A multiuser operating system allows many users to share processing time on a powerful central computer from different terminals. The operating system accomplishes this by rapidly switching between terminals, each of which receives a limited amount of processor time on the central computer. The operating system changes among terminals so quickly that each user seems to have continuous access to the central computer. If there are many users on a system like this, the time it takes the central computer to reply can become more noticeable.

1. **Real Time Operating System**

A real time operating system (RTOS) is a special-purpose operating system used on computers with strict time limits for any work to be done. It is widely used in those systems where the results of the computations are used to influence the process during execution. Whenever an external event occurs to the computer, it is transmitted to the computer with the help of a special sensor used to monitor the event. The sensor generates a signal that is translated by the operating system as an interrupt. When it detects an interrupt, the operating system requests a specific procedure or set of processes to serve the interrupt. With an RTOS, the time taken for processing is measured in tenths of seconds. This system is designed in a way that it is time-bound and has fixed timelines to serve.

1. **Multi-Processor System**

In operating systems, to improve the performance of more than one CPU can be used within one computer system called Multiprocessor operating system. Multiple CPUs are interconnected so that a job can be divided among them for faster execution. When a job finishes, results from all CPUs are collected and compiled to give the final output. Jobs needed to share main memory and they may also share other system resources among themselves. Multiple CPUs can also be used to run multiple jobs simultaneously.

**For Example: UNIX Operating system is one of the most widely used multiprocessing systems.**

1. **Distributed Operating System**

A distributed operating system is an essential type of operating system. Distributed systems use many central processors to serve multiple real-time applications and users. As a result, data processing jobs are distributed between the processors. It connects multiple computers via a single communication channel. Furthermore, each of these systems has its own processor and memory. Additionally, these CPUs communicate via high-speed buses or telephone lines. Individual systems that communicate via a single channel are regarded as a single entity. They're also known as loosely coupled systems.

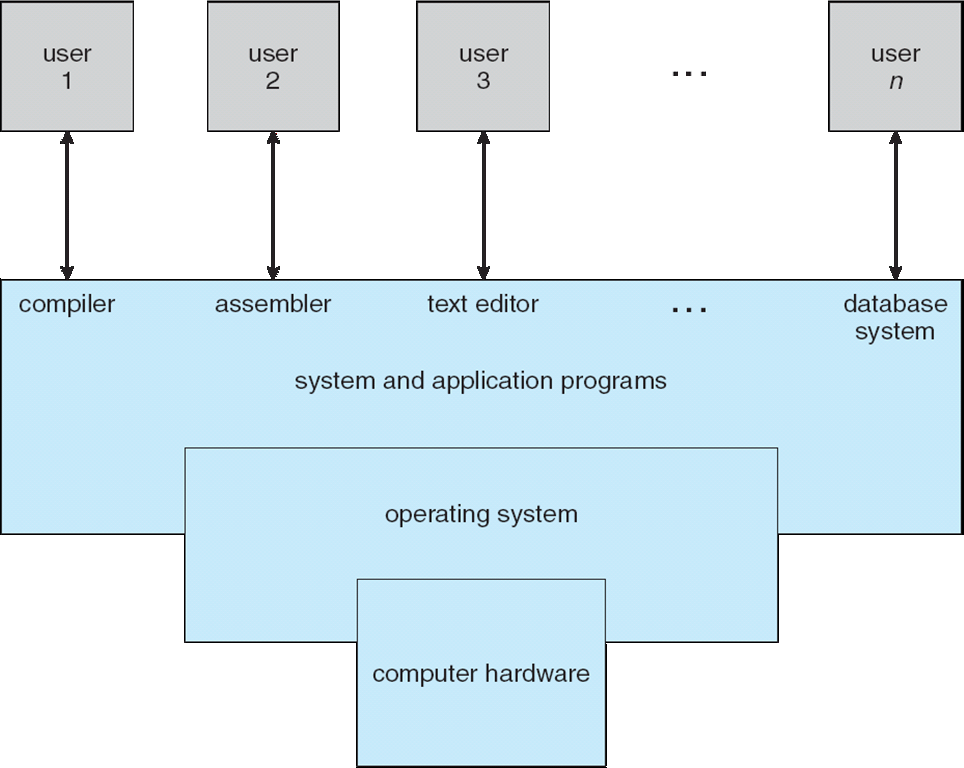
**Computer System Structure:**

* Computer system can be divided into four components:
  + Hardware – provides basic computing resources.
    - CPU, memory, I/O devices
  + Operating system
    - Controls and coordinates use of hardware among various applications and users
  + Application programs – define the ways in which the system resources are used to solve the computing problems of the users
    - Word processors, compilers, web browsers, database systems, video games
  + Users
    - People, machines, other computers.

**What is an Operating System?**

* A program that acts as an intermediary between a user of a computer and the computer hardware
* Operating system goals:
  + Execute user programs and make solving user problems easier.
  + Make the computer system convenient to use.
  + Use the computer hardware in an efficient manner.

**Four Components of a Computer System:**



What Operating Systems Do:

* The operating system controls the hardware and coordinates its use among the various application programs for the various users.
* We can also view a computer system as consisting of hardware, software, and data.
* The operating system provides the means for proper use of these resources in the operation of the computer system.
* An operating system is similar to a government. Like a government, it performs no useful function by itself. It simply provides an environment within which other programs can do useful work.
* To understand more fully the operating system's role, we explore operating systems from two viewpoints:
  + The user
  + The system.

User View:

The user's view of the computer varies according to the interface being used

* Single user computers (e.g., PC, workstations). Such systems are designed for one user to monopolize its resources. The goal is to maximize the work (or play) that the user is performing. the operating system is designed mostly for ease of use and good performance.
* Multi user computers (e.g., mainframes, computing servers). These users share resources and may exchange information. The operating system in such cases is designed to maximize resource utilization -- to assure that all available CPU time, memory, and I/O are used efficiently and that no individual users takes more than their air share.
* Handheld computers (e.g., smartphones and tablets). The user interface for mobile computers generally features a touch screen. The systems are resource poor, optimized for usability and battery life.
* Embedded computers (e.g., computers in home devices and automobiles) The user interface may have numeric keypads and may turn indicator lights on or off to show status. The operating systems are designed primarily to run without user intervention.

System View:

From the computer's point of view, the operating system is the program most intimately involved with the hardware. There are two different views:

* The operating system is a resource allocator
  + Manages all resources
  + Decides between conflicting requests for efficient and fair resource use
* The operating systems is a control program
  + Controls execution of programs to prevent errors and improper use of the computer

Defining Operating System:

No universally accepted definition of what an OS:

* Operating systems exist to offer a reasonable way to solve the problem of creating a usable computing system.
* The fundamental goal of computer systems is to execute user programs and to make solving user problems easier.
* Since bare hardware alone is not particularly easy to use, application programs are developed.
  + These programs require certain common operations, such as those controlling the I/O devices.
  + The common functions of controlling and allocating resources are brought together into one piece of software: the operating system.
  + A simple viewpoint is that it includes everything a vendor ships when you order the operating system. The features that are included vary greatly across systems:
  + Some systems take up less than a megabyte of space and lack even a full-screen editor,
  + Some systems require gigabytes of space and are based entirely on graphical windowing systems.
  + A more common definition, and the one that we usually follow, is that the operating system is the one program running at all times on the computer -- usually called the kernel.
  + Along with the kernel, there are two other types of programs:
  + System programs, which are associated with the operating system but are not necessarily part of the kernel.
  + Application programs, which include all programs not associated with the operation of the system.
  + The emergence of mobile devices have resulted in an increase in the number of features that constituting the operating system.
  + Mobile operating systems often include not only a core kernel but also middleware -- a set of software frameworks that provide additional services to application developers.
  + For example, each of the two most prominent mobile operating systems -- Apple's iOS and Google's Android -- feature a core kernel along with middleware that supports databases, multimedia, and graphics (to name only a few).

**Text & Reference books:**

1. Operating Systems: Three Easy Pieces, Remzi H. Arpaci-Dusseau and Andrea C. Arpaci- Dusseau, Arpaci-Dusseau Books, May, (2014).
2. Silberschatz, A., Galvin, P.B. and Gagne, G., 2018. Operating system concepts essentials. John Wiley & Sons, Inc.