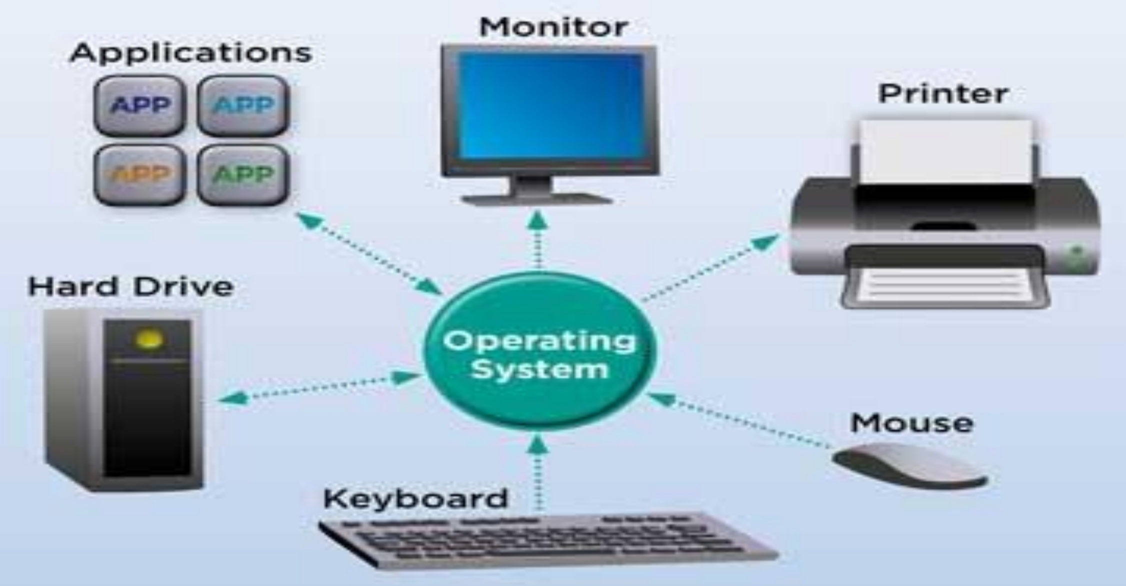
# Unit -1

## What is an Operating System

An **Operating System** is a System software that manages all the resources of the computing device.

- Acts as an interface between the software and different parts of the computer or the computer hardware.
- Manages the overall resources and operations of the computer.
- Controls and monitors the execution of all other programs that reside in the computer, which also includes application programs and other system software of the computer.
- Examples of Operating Systems are Windows, Linux, macOS, Android, iOS, etc.



### Advantages of Operating System

- Operating System manages external and internal devices for example, printers, scanners, and other.
- Operating System provides interfaces and drivers for proper communication between system and hardware devices.
- Allows multiple applications to run simultaneously.
- Manages the execution of processes, ensuring that the system remains responsive.
- Organizes and manages files on storage devices.
- Operating system allocates resources to various applications and ensures their efficient utilization.

### Disadvantages of Operating System

- If an error occurred in your operating system, then there may be a chance that your data may not be recovered therefore always have a backup of your data.
- Threats and viruses can attack our operating system at any time, making it challenging for the OS to keep the system protected from these dangers.
- For learning about new operating system can be a time-consuming and challenging, Specially for those who using particular Operating system for example switching from Windows OS to Linux is difficult.
- Keeping an operating system up-to-date requires regular maintenance, which can be time-consuming.
- Operating systems consume system resources, including CPU, memory, and storage, which can affect the performance of other applications.

## History of Operating System

An operating system is a type of software that acts as an interface between the user and the hardware. It is responsible for handling various critical functions of the computer and utilizing resources very efficiently so the operating system is also known as a resource manager. The operating system also acts like a government because just as the government has authority over everything, similarly the operating system has authority over all resources. Various tasks that are handled by OS are file management, task management, garbage management, memory management, process management, disk management, I/O management, peripherals management, etc.

**Generations of Operating Systems**

- **1940s-1950s: Early Beginnings**
  - Computers operated without operating systems (OS).
  - Programs were manually loaded and run, one at a time.
  - The first operating system was introduced in 1956. It was a batch processing system GM-NAA I/O (1956) that automated job handling.
- **1960s: Multiprogramming and Timesharing**
  - Introduction of multiprogramming to utilize CPU efficiently.
  - Timesharing systems, like CTSS (1961) and Multics (1969), allowed multiple users to interact with a single system.
- **1970s: Unix and Personal Computers**
  - Unix (1971) revolutionized OS design with simplicity, portability, and multitasking.
  - Personal computers emerged, leading to simpler OSs like CP/M (1974) and PC-DOS (1981).
- **1980s: GUI and Networking**
  - Graphical User Interfaces (GUIs) gained popularity with systems like Apple Macintosh (1984) and Microsoft Windows (1985).
  - Networking features, like TCP/IP in Unix, became essential.
- **1990s: Linux and Advanced GUIs**
  - Linux (1991) introduced open-source development.
  - Windows and Mac OS refined GUIs and gained widespread adoption.
- **2000s-Present: Mobility and Cloud**
  - Mobile OSs like iOS (2007) and Android (2008) dominate.
  - Cloud-based and virtualization technologies reshape computing, with OSs like Windows Server and Linux driving innovation.
- **AI Integration – (Ongoing)**
  With the growth of time, Artificial intelligence came into picture. Operating system integrates features of AI technology like Siri, Google Assistant, and Alexa and became more powerful and efficient in many way. These AI features with operating system create a entire new feature like voice commands, predictive text, and personalized recommendations.

Note: The above mentioned OS basically tells how the OS evolved with the time by adding new features but it doesn't mean that only new generation OS are in use and previously OS system are not in use, according to the need, all these OS are still used in software industry.

## Functions of Operating System

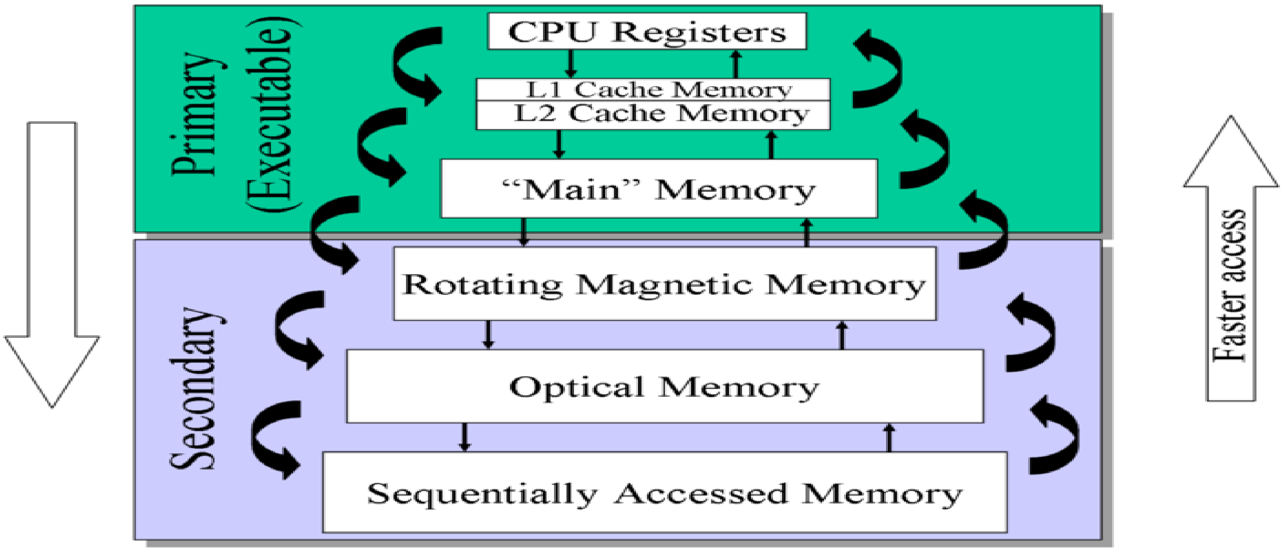
An Operating System acts as a communication interface between the user and computer hardware. Its purpose is to provide a platform on which a user can execute programs conveniently and efficiently. An operating system is software that manages the allocation of Computer Hardware. The coordination of the hardware must be appropriate to ensure the computer system's correct operation and to prevent user programs from interfering with it. The main goal of the Operating System is to make the computer environment more convenient to use and the Secondary goal is to use the resources most efficiently. In this article we will see functions of operating system in detail.

1. **Memory Management**
2. **Processor Management**
3. **Device Management**
4. **File Management**
5. **I/O Management**

### 1. Memory Management

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. An operating system manages the allocation and deallocation of memory to various processes and ensures that the other process does not consume the memory allocated to one process. An Operating System performs the following activities for Memory Management:

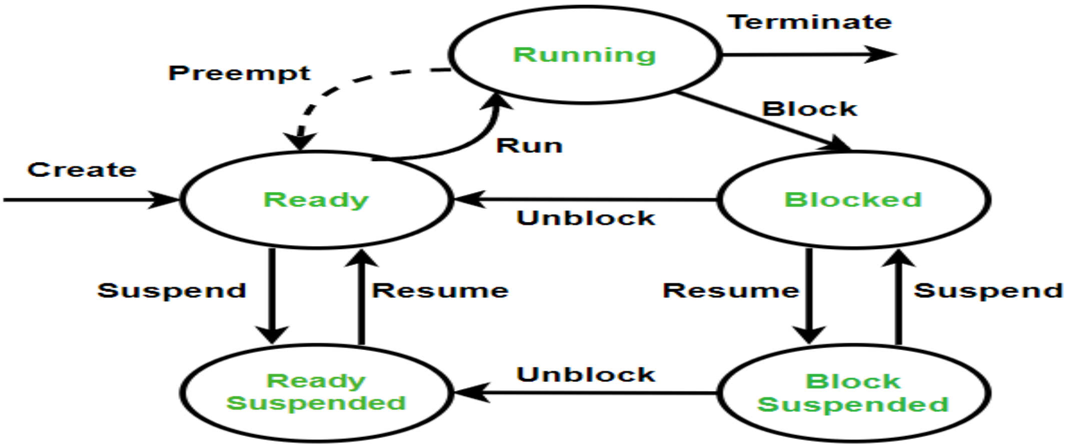
- It keeps track of primary memory, i.e., which bytes of memory are used by which user program. The memory addresses that have already been allocated and the memory addresses of the memory that has not yet been used.
- In multiprogramming, the OS decides the order in which processes are granted memory access, 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. Processor 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.

An operating system manages the processor's work by allocating various jobs to it and ensuring that each process receives enough time from the processor to function properly. 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 longer required.



### 3. Device Management

An OS manages device communication via its respective drivers. It performs the following activities for device management.

- Keeps track of all devices connected to the system. Designates a program responsible for every device known as the Input/output controller.
- Decide which process gets access to a certain device and for how long.
- Allocates devices effectively and efficiently. Deal locates devices when they are no longer required.
- There are various input and output devices. An OS controls the working of these input-output devices.
- It receives the requests from these devices, performs a specific task, and communicates back to the requesting process.

### 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 following file management activities. It keeps track of where information is stored, user access settings, the status of every file, and more. These facilities are collectively known as the file system. An OS keeps track of information regarding the creation, deletion, transfer, copy, and storage of files in an organized way. It also maintains the integrity of the data stored in these files, including the file directory structure, by protecting against unauthorized access.

### 5. I/O Management

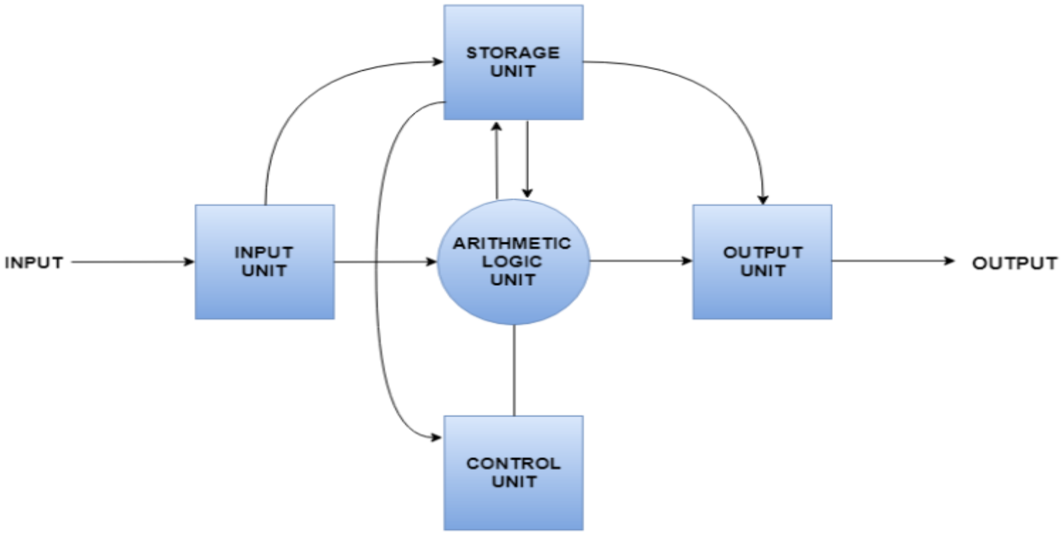
**I/O management** is the important function of operating system refers to how the OS handles **input** and **output** operations between the computer and external devices, such as keyboards, mice, printers, hard drives, and monitors.

### User Interface or Command Interpreter

The user interacts with the computer system through the operating system. Hence OS acts as an interface between the user and the computer hardware. This user interface is offered through a set of commands or a graphical user interface (GUI). Through this interface, the user makes interacts with the applications and the machine hardware.

## Computer System Architecture

**A computer system is basically a machine that simplifies complicated tasks. It should maximize performance and reduce costs as well as power consumption. The different components in the Computer System Architecture are Input Unit, Output Unit, Storage Unit,** Arithmetic Logic Unit**,** Control Unit **etc.**



**The input data travels from input unit to ALU. Similarly, the computed data travels from ALU to output unit. The data constantly moves from storage unit to ALU and back again. This is because stored data is computed on before being stored again. The control unit controls all the other units as well as their data.**

- **Input Unit**

  The input unit provides data to the computer system from the outside. So, basically it links the external environment with the computer. It takes data from the input devices, converts it into machine language and then loads it into the computer system. Keyboard, mouse etc. are the most commonly used input devices.
- **Output Unit**

  The output unit provides the results of computer process to the users i.e it links the computer with the external environment. Most of the output data is the form of audio or video. The different output devices are monitors, printers, speakers, headphones etc.
- **Storage Unit**

  Storage unit contains many computer components that are used to store data. It is traditionally divided into **primary storage and secondary storage**. Primary storage is also known as the main memory and is the memory directly accessible by the **CPU**. Secondary or external storage is not directly accessible by the CPU. The data from secondary storage needs to be brought into the primary storage before the CPU can use it. Secondary storage contains a large amount of data permanently.
- **Arithmetic Logic Unit**

  All the calculations related to the computer system are performed by the arithmetic logic unit. It can perform operations like addition, subtraction, multiplication, division etc. The control unit transfers data from storage unit to arithmetic logic unit when calculations need to be performed. The arithmetic logic unit and the control unit together form the central processing unit.
- **Control Unit**

  This unit controls all the other units of the computer system and so is known as its central nervous system. It transfers data throughout the computer as required including from storage unit to central processing unit and vice versa. The control unit also dictates how the memory, input output devices, arithmetic logic unit etc. should behave.

## Operating System Services

An operating system is software that acts as an intermediary between the user and computer hardware. It is a program with the help of which we are able to run various applications. It is the one program that is running all the time. Every computer must have an operating system to smoothly execute other programs.
The OS coordinates the use of the hardware and application programs for various users. It provides a platform for other application programs to work. The operating system is a set of special programs that run on a computer system that allows it to work properly. It controls input-output devices, execution of programs, managing files, etc.

**Services of Operating System**

- Program execution
- Input Output Operations
- Communication between Process
- File Management
- Memory Management
- Process Management

- Security and Privacy
- Resource Management
- User Interface
- Networking
- Error handling
- Time Management

**Program Execution**

It is the Operating System that manages how a program is going to be executed. It loads the program into the memory after which it is executed. The order in which they are executed depends on the CPU Scheduling Algorithms. A few are FCFS, SJF, etc. When the program is in execution, the Operating System also handles deadlock i.e. no two processes come for execution at the same time. The Operating System is responsible for the smooth execution of both user and system programs. The Operating System utilizes various resources available for the efficient running of all types of functionalities.

**Input Output Operations**

Operating System manages the input-output operations and establishes communication between the user and device drivers. Device drivers are software that is associated with hardware that is being managed by the OS so that the sync between the devices works properly. It also provides access to input-output devices to a program when needed.

**Communication Between Processes**

The Operating system manages the communication between processes. Communication between processes includes data transfer among them. If the processes are not on the same computer but connected through a computer network, then also their communication is managed by the Operating System itself.

**File Management**

The operating system helps in managing files also. If a program needs access to a file, it is the operating system that grants access. These permissions include read-only, read-write, etc. It also provides a platform for the user to create, and delete files. The Operating System is responsible for making decisions regarding the storage of all types of data or files, i.e, floppy disk/hard disk/pen drive, etc. The Operating System decides how the data should be manipulated and stored.

**Memory Management**

Let’s understand memory management by OS in simple way. Imagine a cricket team with limited number of player . The team manager (OS) decide whether the upcoming player will be in playing 11 ,playing 15 or will not be included in team , based on his performance . In the same way, OS first check whether the upcoming program fulfil all requirement to get memory space or not ,if all things good, it checks how much memory space will be sufficient for program and then load the program into memory at certain location. And thus , it prevents program from using unnecessary memory.

**Process Management**

Let’s understand the process management in unique way. Imagine, our kitchen stove as the (CPU) where all cooking(execution) is really happen and chef as the (OS) who uses kitchen-stove(CPU) to cook different dishes(program). The chef(OS) has to cook different dishes(programs) so he ensure that any particular dish(program) does not take long time(unnecessary time) and all dishes(programs) gets a chance to cooked(execution) .The chef(OS) basically scheduled time for all dishes(programs) to run kitchen(all the system) smoothly and thus cooked(execute) all the different dishes(programs) efficiently.

**Security and Privacy**

- **Security :** OS keep our computer safe from an unauthorized user by adding security layer to it. Basically, Security is nothing but just a layer of protection which protect computer from bad guys like viruses and hackers. OS provide us defenses like firewalls and anti-virus software and ensure good safety of computer and personal information.
- **Privacy :** OS give us facility to keep our essential information hidden like having a lock on our door, where only you can enter and other are not allowed . Basically , it respect our secrets and provide us facility to keep it safe.

**Resource Management**

System resources are shared between various processes. It is the Operating system that manages resource sharing. It also manages the CPU time among processes using CPU Scheduling Algorithms. It also helps in the memory management of the system. It also controls input-output devices. The OS also ensures the proper use of all the resources available by deciding which resource to be used by whom.

### User Interface

User interface is essential and all operating systems provide it. Users either interacts with the operating system through the command-line interface or graphical user interface or GUI. The command interpreter executes the next user-specified command.

A GUI offers the user a mouse-based window and menu system as an interface.

### Networking

This service enables communication between devices on a network, such as connecting to the internet, sending and receiving data packets, and managing network connections.

### Error Handling

The Operating System also handles the error occurring in the CPU, in Input-Output devices, etc. It also ensures that an error does not occur frequently and fixes the errors. It also prevents the process from coming to a deadlock. It also looks for any type of error or bugs that can occur while any task. The well-secured OS sometimes also acts as a countermeasure for preventing any sort of breach of the Computer System from any external source and probably handling them.

### Conclusion

In conclusion, operating system services are essential for managing computer resources, ensuring security and stability, supporting multitasking, providing control over system operations, enabling input/output operations, and facilitating networking and communication. These services allow users and applications to interact efficiently with the hardware and software, making the overall computing experience smooth and effective.

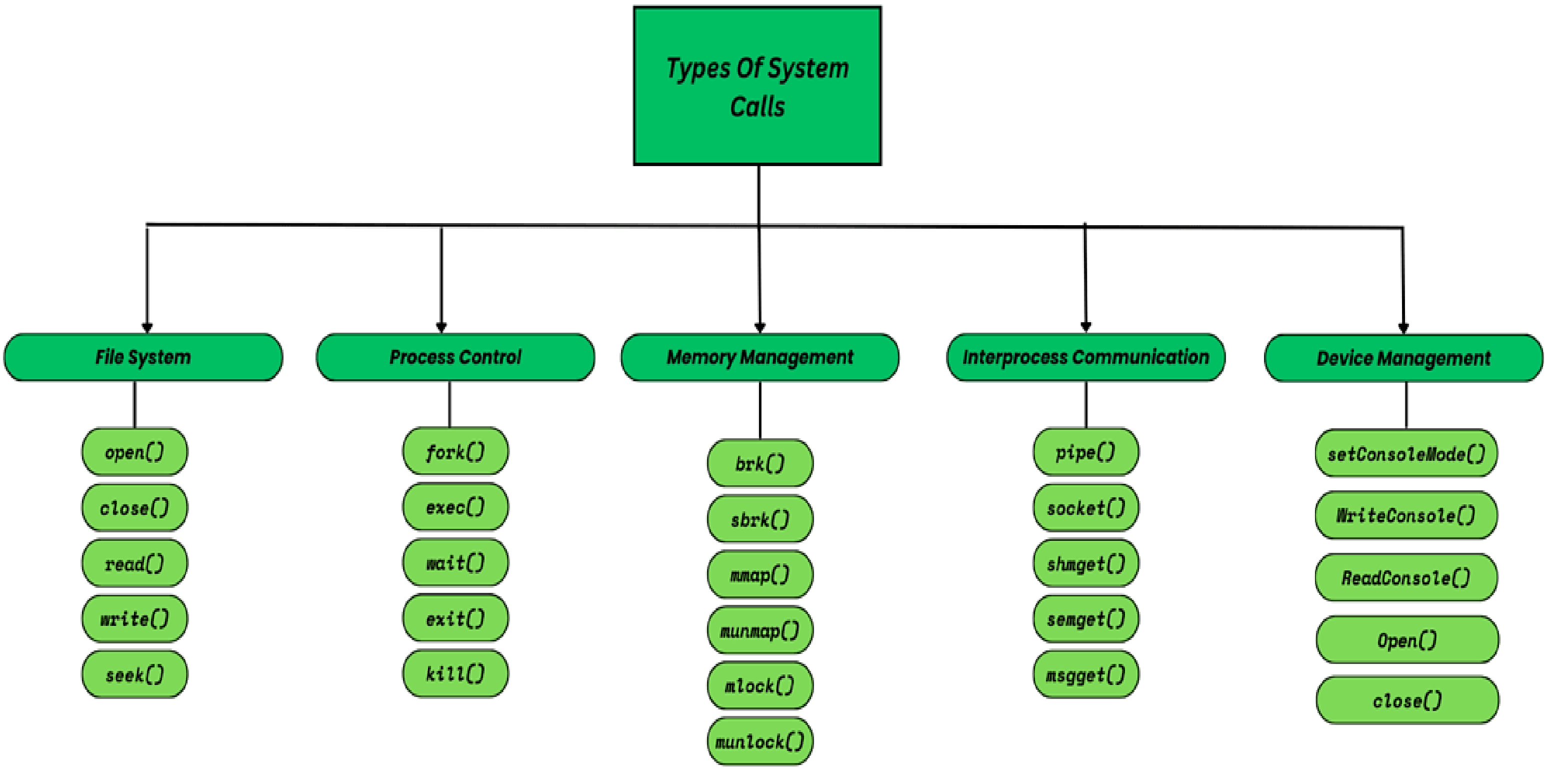
## Different Types of System Calls in OS

System calls are interfaces provisioned by the operating system to allow user-level applications to interact with low-level hardware components & make use of all the services provided by the kernel, which is a core component and the heart of an operating system that manages all the hardware and the services provided by the OS.

These system calls are essential for every process to interact with the kernel and properly use the services provided by it. System calls are an interface between a process and the operating system. And they're the only way to switch from user mode to kernel mode.

## Types of System Calls

Services provided by an OS are typically related to any kind of operation that a user program can perform like creation, termination, forking, moving, communication, etc. Similar types of operations are grouped into one single system call category. System calls are classified into the following categories:



**File System Operations**

These system calls are made while working with files in OS, File manipulation operations such as creation, deletion, termination etc.

- **open():** Opens a file for reading or writing. A file could be of any type like text file, audio file etc.
- **read():** Reads data from a file. Just after the file is opened through open() system call, then if some process want to read the data from a file, then it will make a read() system call.
- **write():** Writes data to a file. Wheneve the user makes any kind of modification in a file and saves it, that's when this is called.
- **close():** Closes a previously opened file.

**Process Control**

These types of system calls deal with process creation, process termination, process allocation, deallocation etc. Basically manages all the process that are a part of OS.

- **fork():** Creates a new process (child) by duplicating the current process (parent). This call is made when a process makes a copy of itself and the parent process is halted temporarily until the child process finishes its execution.
- **exec():** Loads and runs a new program in the current process and replaces the current process with a new process. All the data such as stack, register, heap memory everything is replaced by a new process and this is known as overlay. For example, when you execute a java byte code using command - java "filename". Then in the background, exec() call will be made to execute the java file and JVM will also be executed.
- **wait():** The primary purpose of this call is to ensure that the parent process doesn't proceed further with its execution until all its child processes have finished their execution. This call is made when one or more child processes are forked.
- **exit()**: It simply terminates the current process.

**Memory Management**

These types of system calls deals with memory allocation, deallocation & dynamically changing the size of a memory allocated to a process. In short, the overall management of memory is done by making these system calls.

- **brk():** Changes the data segment size for a process in HEAP Memory. It takes an address as argument to define the end of the heap and explicitly sets the size of HEAP.
- **sbrk():** This call is also for memory management in heap, it also takes an argument as an integer (+ve or -ve) specifying whether to increase or decrease the size respectively.
- **mmap(): Memory** Map - It basically maps a file or device into main memory and further into a process's address space for performing operations. And any changes made in the content of a file will be reflected in the actual file.
- **munmap():** Unmaps a memory-mapped file from a process's address space and out of main memory
- **mlock() and unlock():** memory lock defines a mechanism through which certain pages stay in memory and are not swapped out to the swap space in the disk. This could be done to avoid page faults. Memory unlock is the opposite of lock, it releases the lock previously acquired on pages.

**Device Management**

The device management system calls are used to interact with various peripherial devices attached to the PC or even the management of the current device.

- **SetConsoleMode():** This call is made to set the mode of console (input or output). It allows a process to control various console modes. In windows, it is used to control the behaviour of command line.
- **WriteConsole():** It allows us to write data on console screen.
- **ReadConsole():** It allows us to read data from console screen (if any arguments are provided).
- **open():** This call is made whenever a device or a file is opened. A unique file descriptor is created to maintain the control access to the opened file or device.
- **close():** This call is made when the system or the user closes the file or device.

## Importance of System Calls

- **Efficient Resource Management:** System Calls help your computer manage its resources efficiently. They allocate and manage memory so programs run smoothly without using up too many resources. This is important for multitasking and overall performance.
- **Security and Isolation:** System Calls ensure that one program cannot interfere with or access the memory of another program. This enhances the security and stability of your device.
- **Multitasking Capabilities:** System Calls support multitasking, allowing multiple programs to run simultaneously. This improves productivity and makes it easy to switch between applications.
- **Enhanced Control:** System Calls provide a high level of control over your device's operations. They allow you to start and stop processes, manage files, and perform various system-related tasks.
- **Input/Output (I/O) Operations:** System Calls enable communication with input and output devices, such as your keyboard, mouse, and screen. They ensure that these devices work effectively.
- **Networking and Communication:** System Calls facilitate networking and communication between different applications. They make it easy to transfer data over networks, browse the web, send emails, and connect online.

## Introduction to UNIX System

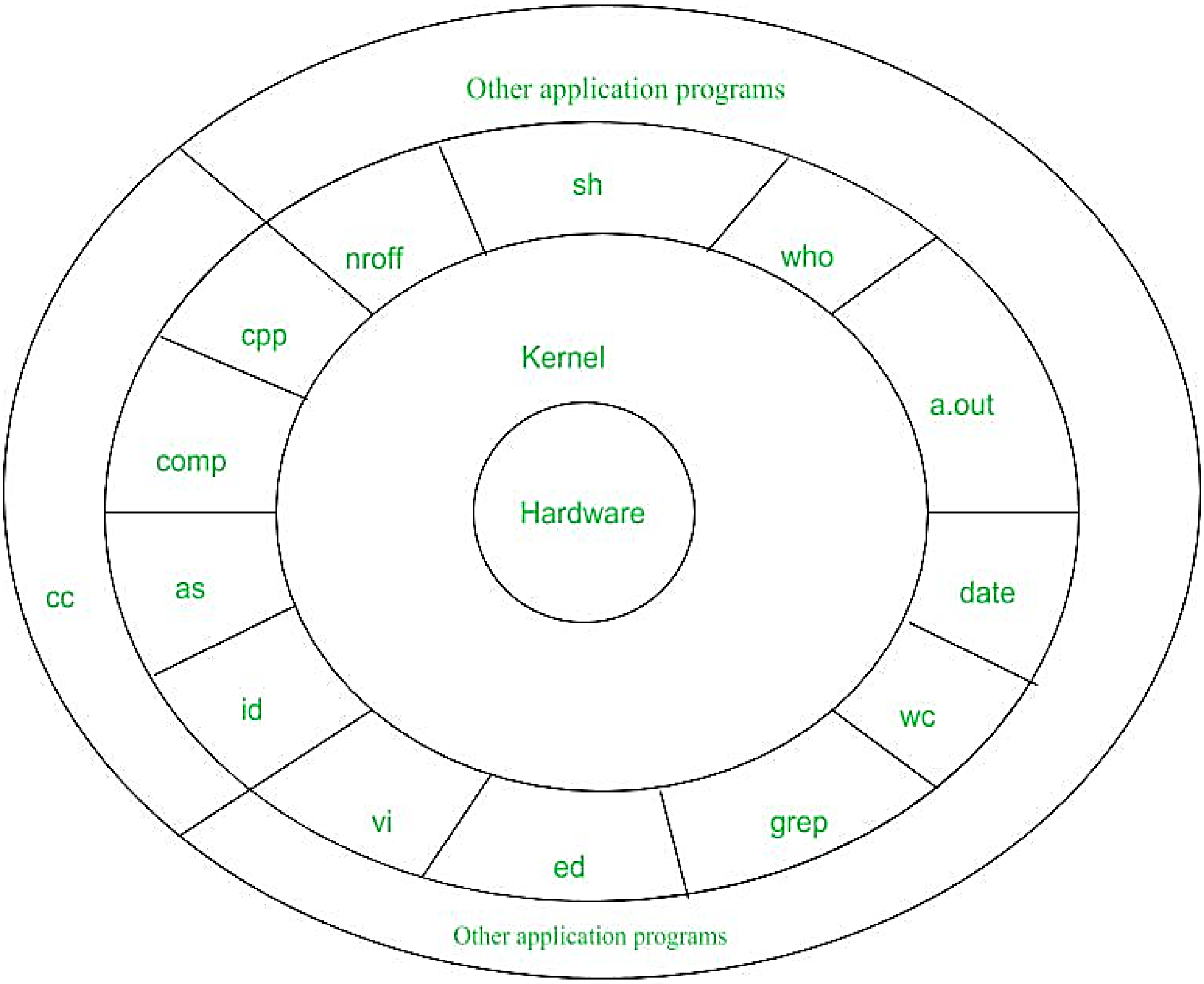
**UNIX** is an innovative or groundbreaking operating system which was developed in the 1970s by Ken Thompson, Dennis Ritchie, and many others at AT&T Laboratories. It is like a backbone for many modern operating systems like Ubuntu, Solaris, Kali Linux, Arch Linux, and also POSIX. Originally, It was designed for developers only, UNIX played a most important role in the development and creation of the software and computing environments. Its distribution to government and academic institutions led to its widespread adoption across various types of hardware components. The core part of the UNIX system lies in its base Kernel, which is integral to its architecture, structure, and key functionality making it the heart of the operating system.

The basic design philosophy of UNIX is to provide simple, powerful tools that can be combined to perform complex tasks. It features a command-line interface that allows users to interact with the system through a series of commands, rather than through a graphical user interface (GUI).

### Some of the Key Features of UNIX Include

1. **Multiuser support:** UNIX allows multiple users to simultaneously access the same system and share resources.
2. **Multitasking:** UNIX is capable of running multiple processes at the same time.
3. **Shell scripting:** UNIX provides a powerful scripting language that allows users to automate tasks.
4. **Security:** UNIX has a robust security model that includes file permissions, user accounts, and network security features.
5. **Portability:** UNIX can run on a wide variety of hardware platforms, from small embedded systems to large mainframe computers.
6. **Communication:** UNIX supports communication methods using the write command, mail command, etc.
7. **Process Tracking:** UNIX maintains a record of the jobs that the user creates. This function improves system performance by monitoring CPU usage. It also allows you to keep track of how much disk space each user uses, and the use that information to regulate disk space.

Today, UNIX is widely used in enterprise-level computing, scientific research, and web servers. Many modern operating systems, including Linux and macOS, are based on UNIX or its variants.


Other application programs
sh
nroff
who
cpp
Kernel
a.out
comp
Hardware
as
date
cc
id
wc
vi
grep
ed
Other application programs