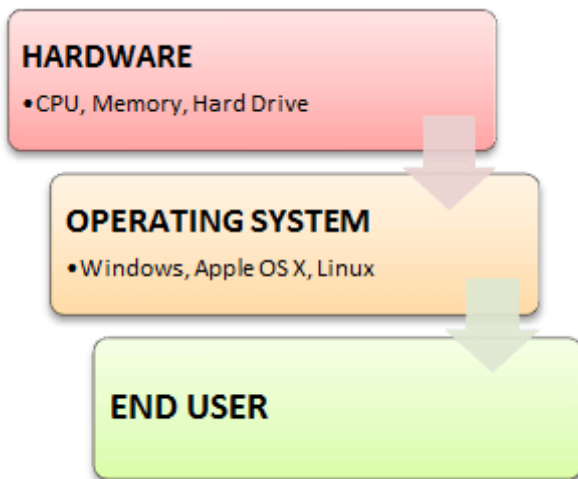# What is an Operating System?

An **Operating System (OS)** is a software that acts as an interface between computer hardware components and the user. Every computer system must have at least one operating system to run other programs. Applications like Browsers, MS Office, Notepad Games, etc., need some environment to run and perform its tasks.

The OS helps you to communicate with the computer without knowing how to speak the computer's language. It is not possible for the user to use any computer or mobile device without having an operating system.

Introduction to Operating System

In this OS tutorial, you will learn:

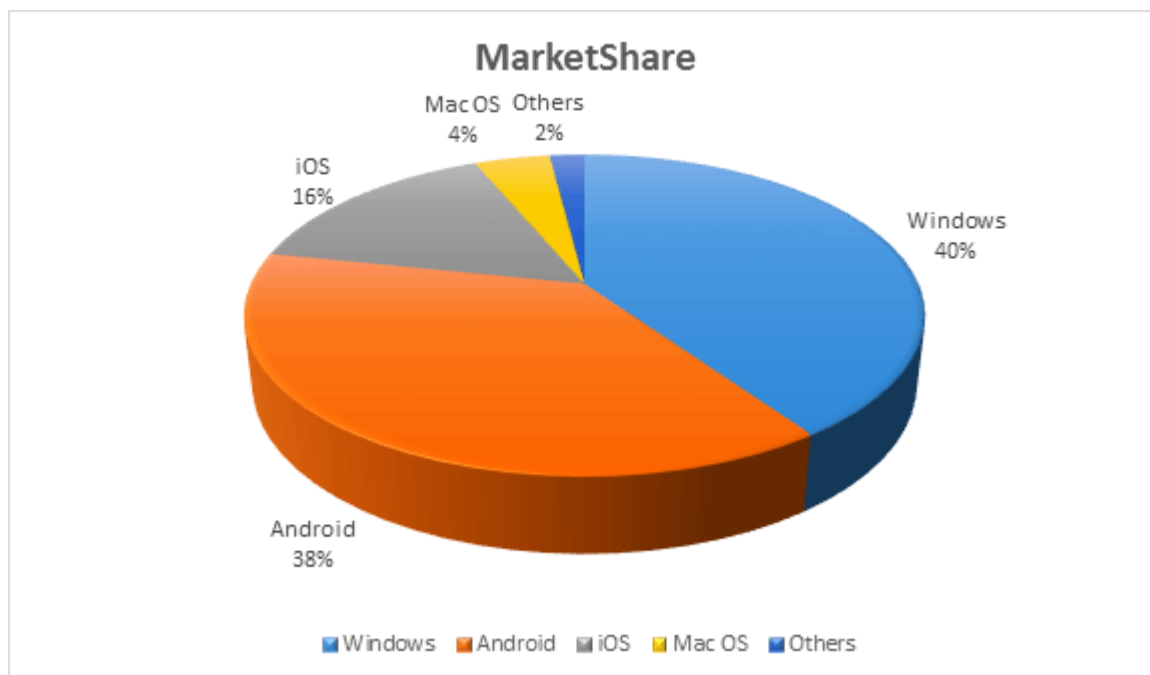- What is an Operating System?
- History Of OS
- Examples of Operating System with Market Share
- Types of Operating System (OS)
- Functions of Operating System
- Features of Operating System (OS)
- Advantage of Operating System
- Disadvantages of Operating System
- What is Kernel in Operating System?
- Features of Kernel
- Types of Kernel
- Difference between Firmware and Operating System

- [Difference between 32-Bit and 64-Bit Operating System](#)

History Of OS

- Operating systems were first developed in the late 1950s to manage tape storage
- The General Motors Research Lab implemented the first OS in the early 1950s for their IBM 701
- In the mid-1960s, operating systems started to use disks
- In the late 1960s, the first version of the Unix OS was developed
- The first OS built by Microsoft was DOS. It was built in 1981 by purchasing the 86-DOS software from a Seattle company
- The present-day popular OS Windows first came to existence in 1985 when a GUI was created and paired with MS-DOS.

Examples of Operating System with Market Share

Market Share of Operating Systems

Following are the Operating System examples with the latest Market Share

OS Name	Share
Windows	40.34
Android	37.95
iOS	15.44
Mac OS	4.34
Linux	0.95
Chrome OS	0.14
Windows Phone OS	0.06

Types of Operating System (OS)

Following are the popular types of OS (Operating System):
- Batch Operating System
- Multitasking/Time Sharing OS
- Multiprocessing OS
- Real Time OS
- Distributed OS
- Network OS
- Mobile OS

Batch Operating System

Some computer processes are very lengthy and time-consuming. To speed the same process, a job with a similar type of needs are batched together and run as a group.

The user of a batch operating system never directly interacts with the computer. In this type of OS, every user prepares his or her job on an offline device like a punch card and submit it to the computer operator.

Multi-Tasking/Time-sharing Operating systems

Time-sharing operating system enables people located at a different terminal(shell) to use a single computer system at the same time. The processor time (CPU) which is shared among multiple users is termed as time sharing.

Real time OS

A real time operating system time interval to process and respond to inputs is very small. Examples: Military Software Systems, Space Software Systems are the Real time OS example.

Distributed Operating System

Distributed systems use many processors located in different machines to provide very fast computation to its users.

Network Operating System

Network Operating System runs on a server. It provides the capability to serve to manage data, user, groups, security, application, and other networking functions.

Mobile OS

Mobile operating systems are those OS which are especially designed to power smartphones, tablets, and wearables devices.

Some most famous mobile operating systems are Android and iOS, but others include BlackBerry, Web, and watchOS.

Functions of Operating System

Some typical operating system functions may include managing memory, files, processes, I/O system & devices, security, etc.

Below are the main functions of Operating System:
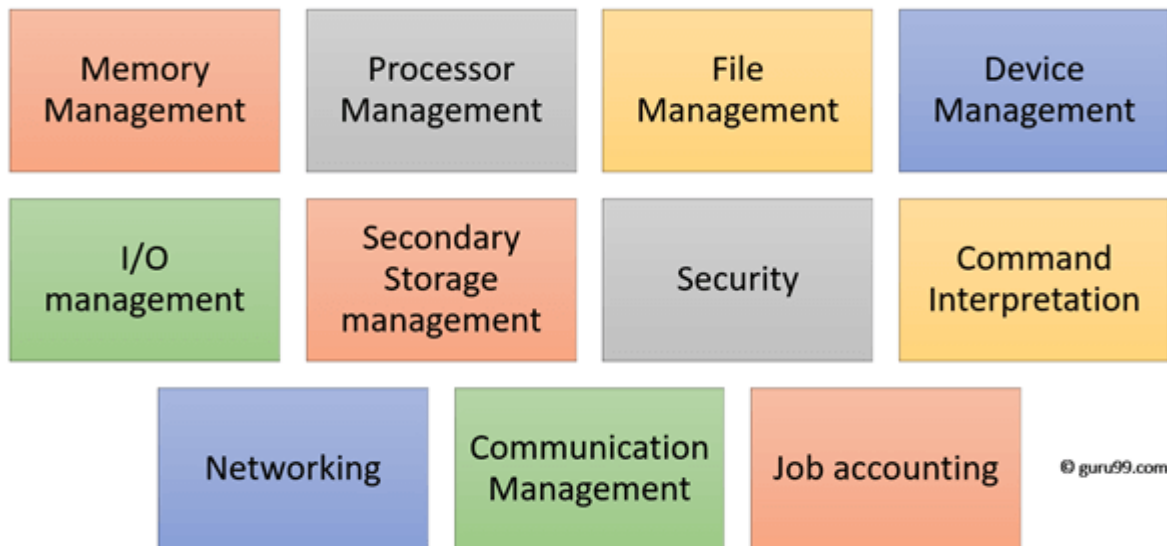
Functions of Operating System

In an operating system software performs each of the function:

1. **Process management**: Process management helps OS to create and delete processes. It also provides mechanisms for synchronisation and communication among processes.

2. **Memory management:** Memory management module performs the task of allocation and deallocation of memory space to programs in need of these resources.

3. **File management**: It manages all the file-related activities such as organization storage, retrieval, naming, sharing, and protection of files.

4. **Device Management**: Device management keeps tracks of all devices. This module also responsible for this task is known as the I/O controller. It also performs the task of allocation and de-allocation of the devices.

5. **I/O System Management:** One of the main objects of any OS is to hide the peculiarities of that hardware devices from the user.

6. **Secondary-Storage Management**: Systems have several levels of storage which includes primary storage, secondary storage, and cache storage. Instructions and data must be stored in primary storage or cache so that a running program can reference it.

7. **Security**: Security module protects the data and information of a computer system against malware threat and authorized access.

8. **Command interpretation**: This module is interpreting commands given by the and acting system resources to process that commands.

9. **Networking:** A distributed system is a group of processors which do not share memory, hardware devices, or a clock. The processors communicate with one another through the network.
10. **Job accounting**: Keeping track of time & resources used by various jobs and users.
11. **Communication management**: Coordination and assignment of compilers, interpreters, and another software resource of the various users of the computer systems.

Features of Operating System (OS)

Here is a list important features of OS:

- Protected and supervisor mode
- Allows disk access and file systems Device drivers Networking Security
- Program Execution
- Memory management Virtual Memory Multitasking
- Handling I/O operations
- Manipulation of the file system
- Error Detection and handling
- Resource allocation
- Information and Resource Protection
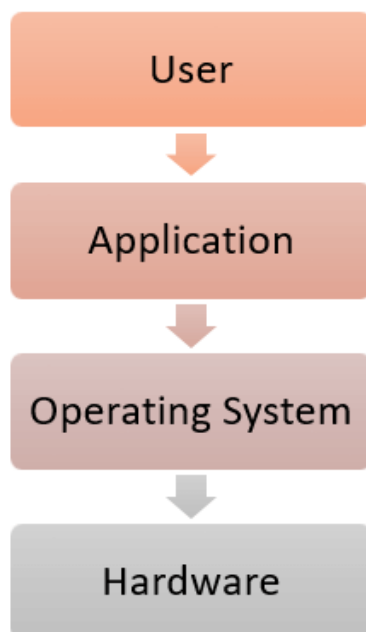
Advantage of Operating System

- Allows you to hide details of hardware by creating an abstraction
- Easy to use with a GUI
- Offers an environment in which a user may execute programs/applications
- The operating system must make sure that the computer system convenient to use
- Operating System acts as an intermediary among applications and the hardware components
- It provides the computer system resources with easy to use format
- Acts as an intermediator between all hardware's and software's of the system

Disadvantages of Operating System

- If any issue occurs in OS, you may lose all the contents which have been stored in your system
- Operating system's software is quite expensive for small size organization which adds burden on them. Example Windows
- It is never entirely secure as a threat can occur at any time

What is Kernel in Operating System?

A Kernel is a computer program that is the heart and core of an Operating System. Since the Operating System has control over the system so, the Kernel also has control over everything in the system. It is the most important part of an Operating System. Whenever a system starts, the Kernel is the first program that is loaded after the bootloader because the Kernel has to handle the rest of the thing of the system for the Operating System. The Kernel remains in the memory until the Operating System is shut-down.

- The Kernel is responsible for low-level tasks such as disk management, memory management, task management, etc.

- It provides an interface between the user and the hardware components of the system.

- When a process makes a request to the Kernel, then it is called System Call.

- A Kernel is provided with a protected Kernel Space which is a separate area of memory and this area is not accessible by other application programs. So, the code of the Kernel is loaded into this protected Kernel Space. Apart from this, the memory used by other applications is called the User Space. As these are two different spaces in the memory, communication between them is a bit slower.

Functions of a Kernel

Following are the functions of a Kernel:

- **Access Computer resource:** A Kernel can access various computer resources like the CPU, I/O devices and other resources. It acts as a bridge between the user and the resources of the system.

- **Resource Management:** It is the duty of a Kernel to share the resources between various processes in such a way that there is uniform access to the resources by every process.

- **Memory Management:** Every process needs some memory space. So, memory must be allocated and deallocated for its execution. All these memory management is done by a Kernel.

- **Device Management:** The peripheral devices connected in the system are used by the processes. So, the allocation of these devices is managed by the Kernel.

Kernel Mode and User Mode

There are certain instructions that need to be executed by Kernel only. So, the CPU executes these instructions in the Kernel Mode only. For example, memory management should be done in Kernel-Mode only. While in the User Mode, the CPU executes the processes that are given by the user in the User Space.

Types of Kernel

In general, there are five types of Kernel. They are: Monolithic Kernels, Microkernel, Hybrid Kernel, Nano Kernel, Exo Kernel

1. Monolithic Kernels

Monolithic Kernels are those Kernels where the user services and the kernel services are implemented in the same memory space i.e. different memory for user services and kernel services are not used in this case. By doing so, the size of the Kernel is increased and this, in turn, increases the size of the Operating System. As there is no separate User Space and Kernel Space, so the execution of the process will be faster in Monolithic Kernels.
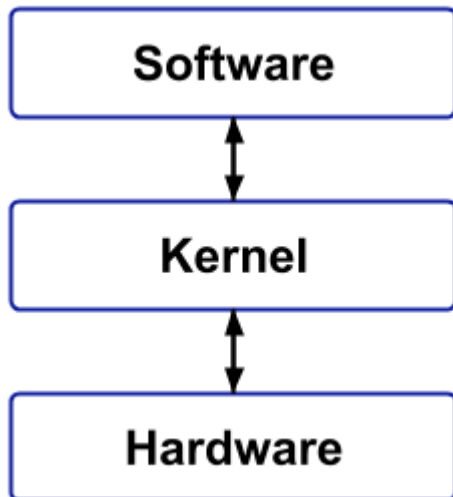
Advantages:

- It provides CPU scheduling, memory scheduling, file management through System calls only.
- Execution of the process is fast because there is no separate memory space for user and kernel.
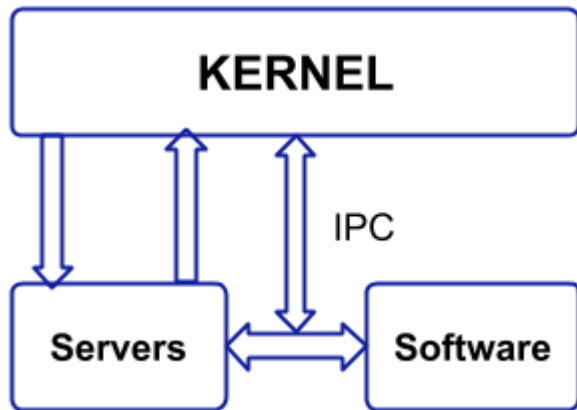
Disadvantages:

- If any service fails, then it leads to system failure.
- If new services are to be added then the entire Operating System needs to be modified.

2. Microkernel

A Microkernel is different from a Monolithic kernel because in a Microkernel, the user services and kernel services are implemented into different spaces i.e. we use User Space and Kernel Space in case of Microkernels. As we are using User Space

and Kernel Space separately, it reduces the size of the Kernel and this, in turn, reduces the size of the Operating System.

As we are using different spaces for user services and kernel service, the communication between application and services is done with the help of message parsing and this, in turn, reduces the speed of execution.

Advantages:

- If new services are to be added then it can be easily added.

Disadvantages:

- Since we are using User Space and Kernel Space separately, the communication between these can reduce the overall execution time.

3. Hybrid Kernel

A Hybrid Kernel is a combination of both Monolithic Kernel and Microkernel. It makes use of the speed of the Monolithic Kernel and the modularity of Microkernel.

Hybrid kernels are micro kernels that have some "non-essential" code in kernel-space in order for the code to run more quickly than it would be in user-space. So, some services such as network stack or filesystem are run in Kernel space to reduce the performance overhead, but still, it runs kernel code as servers in the user-space.

4. Nanokernel

In a Nanokrnel, as the name suggests, the whole code of the kernel is very small i.e. the code executing in the privileged mode of the hardware is very small. The term nanokernel is used to describe a kernel that supports a nanosecond clock resolution.

5. Exokernel

Exokernel is an Operating System kernel that is developed by the MIT parallel and the Distributed Operating Systems group. Here in this type of kernel, the resource protection is separated from the management and this, in turn, results in allowing us to perform application-specific customization.

In the Exokernel, the idea is not to implement all the abstractions. But the idea is to impose as few abstractions as possible and by doing so the abstraction should be used only when needed. So, no force abstraction will be there in Exokernel and this is the feature that makes it different from a Monolithic Kernel and

Microkernel. But the drawback of this is the complex design. The design of the Exokernel is very complex.

Difference between Firmware and Operating System

Below are the Key Differences between Firmware and Operating System:

Firmware	Operating System
Define Firmware: Firmware is one kind of programming that is embedded on a chip in the device which controls that specific device.	Define Operating System: OS provides functionality over and above that which is provided by the firmware.
Firmware is programs that been encoded by the manufacture of the IC or something and cannot be changed.	OS is a program that can be installed by the user and can be changed.
It is stored on non-volatile memory.	OS is stored on the hard drive.

Difference between 32-Bit and 64-Bit Operating System

Below are the Key Differences between 32-Bit and 64-Bit Operating System:

Parameters	32. Bit	64. Bit
Architecture and Software	Allow 32 bit of data processing simultaneously	Allow 64 bit of data processing simultaneously
Compatibility	32-bit applications require 32-bit OS and CPUs.	64-bit applications require a 64-bit OS and CPU.
Systems Available	All versions of Windows 8, Windows 7, Windows Vista, and Windows XP, Linux, etc.	Windows XP Professional, Vista, 7, Mac OS X and Linux.
Memory Limits	32-bit systems are limited to 3.2 GB of RAM.	64-bit systems allow a maximum 17 Billion GB of RAM.

Summary

- What is OS (Operating System definition) and its Types: An operating system is a software which acts as an interface between the end user and computer hardware. Different categories of Operating System in computer and other devices are: Batch Operating System, Multitasking/Time Sharing OS, Multiprocessing OS, Real Time OS, Distributed OS, Network OS & Mobile OS
- Personal Computer Operating Systems were first developed in the late 1950s to manage tape storage
- Explain Operating System working: OS works as an intermediate between the user and computer. It helps the user to communicate with the computer without knowing how to speak the computer's language.
- The kernel is the central component of a computer operating systems. The only job performed by the kernel is to the manage the communication between the software and the hardware

- Two most popular kernels are Monolithic and MicroKernels
- Process, Device, File, I/O, Secondary-Storage, Memory management are various functions of an Operating System