What is OS?

In 1960's definition of an operating system is "The software that controls the hardware". However today due to microcode we need a better definition. We see an operating system as the programs that make the hardware useable. In brief ,an operating system is the set of programs that controls a computer. Controlling the computer involves software at several levels.

An operating system is the primary software that manages all the hardware and other software on a computer. The operating system, also known as an "OS," interfaces with the computer's hardware and provides services that applications can use.

An operating system is a software which acts as an interface between the end user and computer hardware. Every computer must have at least one OS to run other programs.

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.

Operating systems are resource managers. The main resource is computer hardware in the form of processors, storage,input/output devices, communication devices and data. Some of the operating system functions are: implementing the user interface, sharing hardware among users, allowing users to share data among themselves, preventing users from interfering with one another, scheduling resources among users, facilitating input/output, recovering from errors, facilitating parallel operations, and handling network communication.

Some examples of operating systems are UNIX, Mach, MS-DOS, MS-Windows, Windows/NT, Chicago, OS/2,MacOS, VMS, MVS, and VM..

Structure of a Computer System

A Computer System consists of:

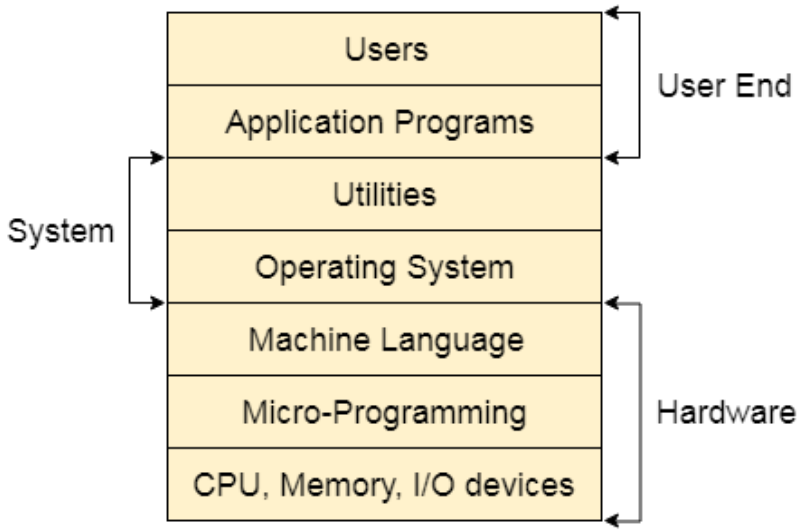
Users (people who are using the computer)

Application Programs (Compilers, Databases, Games, Video player, Browsers, etc.)

System Programs (Shells, Editors, Compilers, etc.)

Operating System (A special program which acts as an interface between user and hardware)

Hardware (CPU, Disks, Memory, etc)



Operating system goals

Efficient use:

An operating system must ensure efficient use of the fundamental computer system resources of memory,CPU, and I/O devices such as disk and printers. Poor efficiency can result if a program does not use a resource allocated to it, e.g if memory or I/O devices allocated to a program remain idle. Such a situation may have a snowballing effect: since the resource is allocated to a program, it is denied to other programs that need it. These programs cannot execute ,hence resource allocated to them also remain idle.in addition the

OS itself consumes some CPU and memory resources during its own operation, and this consumption of resources constitutes an overhead that also reduce the resources available to user programs. To achieve good efficiency , the OS must minimize the waste of resources by programs and also minimizes its own overhead. Efficient Use of resources can be obtained by monitoring use of resources and performing corrective actions when necessary.

User convenience:

Provide convenient methods of using a computer system.

Operating system functions

i) I/O Management:

OS manages I/O devices and makes the I/O process effective. OS accepts the input from the input device, stores it in the main memory, ask the CPU to process it and finally provides the result to the output devices for output.

ii) Command Interpreter:

Command interpreter is one of the part of operating system which reads the commands that user types in at a terminal, interprets them and translate them into a detailed set of instructions that the computer hardware can understand. It varies widely from one OS to other OS. Every OS must provide command interpreter for its operation.

iii) Memory Management:

Memory is the large array of words or bytes, each with its own address. When user requests CPU for read/write operation, OS determines the amount of memory required for the program instructions and data. Then, OS allocates required memory to load the program and data into RAM. When program terminates its memory area is free and the same memory area is allocated for other programs.

iv) Process Management:

OS finds the status of processor and processes, chooses a job, chooses processer in the job, allocates the processor to the process and frees the processor when the process is executed.

v) Time Sharing:

OS manages the time of CPU. The kernel OS checks frequency for other processes requesting CPU time. Time-sharing checks for CPU request from higher priority processes that are made every 10 milliseconds. When two or more processes at the same priority level are competing for the CPU time, CPU time is sliced into segments, defined by time slice and passed from process to process in a round robin fashion, preventing a single process from monopolizing the CPU until it blocks or terminates.

vi) Security:

OS makes sure that only authorized users get access to the computer and its data and the users only do things they are authorized to do.

vii) Deadlock Prevention:

During processing, a situation can arise in which a resource shared by two or more processes cannot continue because the resource required by one process is held by the other. This situation is called deadlock. OS ensures that the above condition do not hold by carefully allocating resources.

viii) Interrupt Handling:

Interrupt is a signal generated from a device or program when they need attention of the CPU. OS determines the type of interrupt and priority of the interrupt, stops the execution process of CPU, preserves the initial state of the CPU, perform the requested operation and brings the CPU at the same state when it was stopped.

ix) Virtual Storage:

If there are programs larger than main memory (RAM) of the computer, OS uses the reserved space in the secondary memory which is termed as virtual memory. It makes the execution of larger program (than RAM) possible but at the same times the operation becomes slower.