Operating System Overview, System Structures

① An OS is an interface between a computer user and computer hardware.

An OS is software that enables applications to interact with a computer hardware. The software that contains the core components of the OS is called kernel.

An OS is software which performs all the basic tasks like file management, memory management, process management, handling IO, and controlling peripheral devices etc.

Ex:- windows OS, Linux, Mac-OS etc.

Definition:-

An OS is the low level software that supports a computers basic functions, such as scheduling tasks and controlling peripherals.

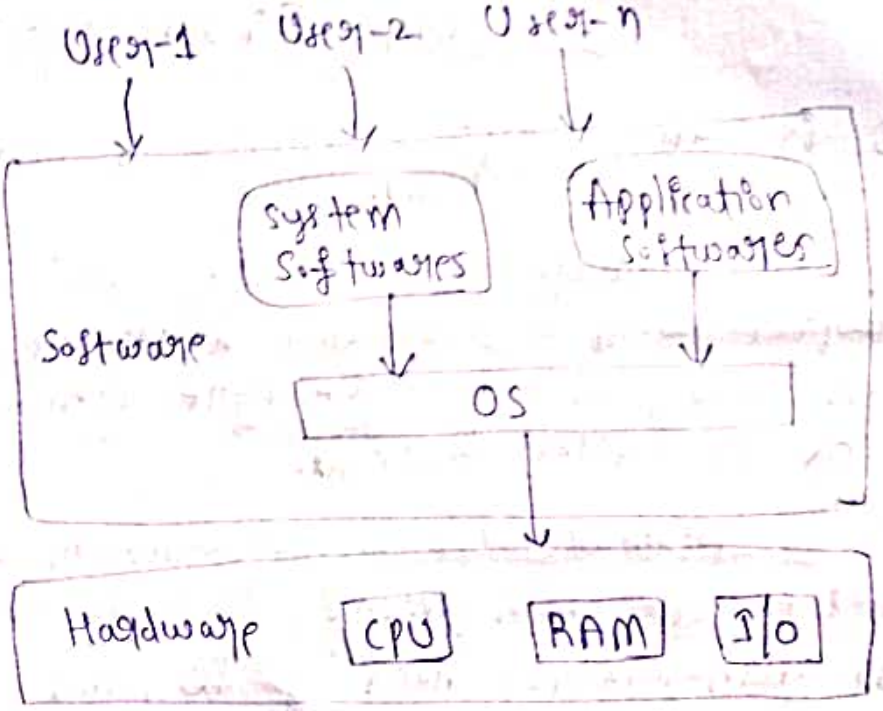
(or)

An OS is a program that acts as an interface b/w the user and the computer hardware and controls the execution of all kinds of programs.

(or)

An OS is system software that manages computer hardware, software resources, and provides common services for computer program

Architecture :-



OS Generations :-

0th generation :-

In this generation ; charles Babbage invented the analytical engine and later john Atanasoff created a computer in 1940. The hardware component of this period was electronic vacuum tube.

1st generation :- (1951-1956)

In this generation, the mono programmed os was developed. Programming language like FORTRAN was developed by John. W. Backus in 1956.

2nd generation :- (1956-1964)

In this generation, the vaccum tubes are replaced by transisters as the hardware component The first os GMOS was developed by the IBM computer.

the Compatible Time Sharing System (CTSS), developed at MIT during the early 1960s.

3rd generation :- (1964-1979)

Hardware technology began to use integrated circuits (ICs).

4th generation :- (1979-present).

The computer technology of the 3rd generation was replaced by very large scale integration (VLSI). Many OS like windows, Linux, MacOS are developed in the 4th generation.

---

② OS Functions :-

The important functions of OS are

(i) Memory Management
(ii) Processor Management
(iii) Device Management
(iv) File Management
(v) Network Management
(vi) Security
(vii) Control over system performance
(viii) Job accounting
(ix) Error detecting

Memory Management :-

It refers to the management of primary memory (or main memory. Main memory is a large array of bytes or words each byte (or word has its own address.

Main memory provides a fast storage that can be accessed directly by the CPU. For a program to be executed, it must be in the main memory.

→ Keeps track of the primary memory.
→ Allocates memory when process request.
→ Deallocates memory when process terminated.

Processor Management:-

In multiprogramming, the OS decides, which process gets executed first and for how much time. This function is called process scheduling.

→ Keeps track of processor and status of the process.
→ Allocates CPU to a process.
→ Deallocates CPU when process no longer available

Device Management:-

An OS manages device communication via drivers.

→ Keeps track of all devices. Program responsible for this task is known as I/O controller.
→ Decides which process gets device & for much time
→ Allocates the device ~~to the process~~.
→ Deallocates the device ~~to the process~~

File Management:-

File system is normally organized into directories for easy navigation and usage.

→ Keeps track of information, location, uses, status etc
→ Decides who gets the resources
→ Allocates the resources to a process.
→ Deallocates the resources.

Security:-
The OS uses a password protection to protect user data, it also prevents unauthorized access to programs and user data.

Control over system performance:-
The OS monitors overall system setup to help in improving the performance and it also records the response time b/w service requests and service response.

Job accounting:-
OS always keep track of time and resources that are used by various tasks and users.

---

(3) OS Operations - Dual-Mode Operation, Timer:-
User application programs can only interact with the system hardware with the help of an OS.
Operating system operations are
(i) Process Management
(ii) Memory Management
(iii) Device Management
(iv) File Management.

Process Management:-

The OS manages the processes, it assigns the processor to process the task, which is termed as process scheduling.

The different types of process scheduling algorithms are

→ FCFS - First Come first served
→ SJF - shortest Job First
→ Priority scheduling
→ Round Robin scheduling.

To handle processes, process management have many queues
ready queue, job queue, device queue.

Memory Management:-

It allocates memory and switches the processes b/w disk and primary memory for execution whenever required.

→ It allocates memory to the processes by using some algorithms like best fit, first fit, and worst fit.
→ OS also deallocates memory from processes when process gets terminated.

Device Management:-

OS handles many IO devices such as mouse, keyboard, disk drive etc.

To handle a specific device, OS can be connected to different device drivers.

All IO devices are accessed by user applications by using device drivers.

File Management :-

To provide a uniform and systematic view of data storage, files are used by the OS.
All the files are mapped to non volatile physical devices in order to protect data loss.
There are 2 ways to access files.
→ Sequential access
→ Direct access.

In sequential access, the files are accessed & processed in compelled order.
Data is accessed one by one.
Ex:- Editors, Compilers etc.

In direct & relative access, the files are accessed in random order for read and write operations.

Trap :-

A trap & an exception is a software-generated interrupt caused either by an error & by a specific request from a user program.

→ The interrupt-driven nature of an OS defines system's general structure.
→ An interrupt service routine is responsible for dealing with the interrupt.

Dual mode Operation :-

There are 2 modes of operation
→ User mode
→ Kernel mode. (Supervisor mode, System mode & privileged mode)

A bit called the mode bit, is added to the hardware of the computer to indicate the current mode.
kernel (0) and user (1).

When the computer system is executing a user application, the system is in user mode.
When the user application requests a service from the OS, it must transition from user to kernel mode to fulfill the request.
The system boot time, the hardware starts in kernel mode.
When a trap (or interrupt occurs, the hardware switches from user mode to kernel mode.
The dual mode of operation provides us for protecting the OS from errant users.
The hardware allows privileged instructions to be executed only in kernel mode.

Timer:-
The OS maintains control over the CPU. We must prevent a user program from getting stuck in an infinite loop, (or not calling system services and never returning control to the OS. To accomplish this task, we can use a timer.
A timer can be set to interrupt the computer after a specified period. The period may be fixed (or variable.

A variable timer is generally implemented by a fixed-rate clock and a counter.
Every second, the timer interrupts and the counter is decremented by 1. The counter is +ve, the control is returned to the user program. If the counter is -ve, the OS terminates the program.

---

④ Types of Computing Environments:-

Computer system uses many devices, arranged in different ways to solve many problems.

The different types of Computing environments are

→ Personal Computing Environment
→ Time Sharing Computing Environment
→ Client Server Computing Environment
→ Distributed Computing Environment
→ Cloud Computing Environment
→ Cluster Computing Environment.

Personal Computing Environment:-

In this environment, there is a single computer system.
All the system processes are available on the computer and executed here.
Ex:- Laptops, Mobiles, Printers, Computer systems, Scanners etc.

Time sharing Computing Environment:-

This environment allows multiple users to share the system simultaneously.

Each user is provided a time slice and the processor switches rapidly among the users according to it.

Client Server Computing Environment:-

In this environment, the client requests a resource and the server provides that resource.

A server may serve multiple clients at the same time while a client is in contact with only one server.

Both the client and server usually communicate via a computer network.

Distributed Computing Environment:-

This environment contains multiple nodes that are physically seperate but linked together using the network.

All the nodes in this system communicate with each other and handle processes in tandem.

Cloud Computing Environment:-

Here computing is not done in individual computer rather it is computed in cloud of computers where all required resources are provided by cloud vendor. This environment primarily comprised of three services.

→ SaaS - Software as a service
→ Iaas - Infrastructure as a service
→ Paas → Platform as a service.

Cluster Computing Environment:-

This environment is similar to parallel computing environment as they both have multiple CPUs. Here cluster performs task. where cluster is a set of loosely (or) tightly connected computers that work together.

---

⑤ Open-Source Operating System:-

The term "open source" refers to computer software (or) applications where the owners (or) copyright holders enable the users to use, see, and edit the products source code.

The source code of an open source OS is publicly visible and editable.
The OSs such as Apples IOS, Microsoft Windows, Mac OS are closed source operating systems.

Linux is an example of most popular open source operating system. The user may modify (or) change those codes and develop new applications according to the user requirement.
Ex:- Linux, Open Solaris, Open BDS, Minix etc.
In 1997, the first open source software was released.

Open source OS works similar to a closed OS, except that the user may modify the source code of the program.

Linux kernel:-
Linux kernel was developed by Linus Torvalds. It offers essential functions required for an OS, such as data concellation, memory processing, and interactions with computer hardware.

Linux Lite:-
Linux lite is another feree and open source OS that can run on lower-end hardware. It is a light weight operating system designed to help users who are unfamiliar with Linux based OSs.

Fedora:-
Fedora is another popular Linux-based OS, and it is widely considered the best open-source OS after Ubuntu.
React OS, Solus, Chromie OS

Advantages of Open Source OS:-
(i) Reliable and Efficient
(ii) Cost efficient
(iii) Flexibility

Disadvantages:-
(i) Complicated
(ii) Security Risk.
(iii) No support

(4) OS Services :-

An OS provides services to both the users and to the programs.

It provides the services to the users to execute programs in a convenient manner.

The services of the OS are

(i) Program Execution
(ii) IO operations
(iii) File system Manipulation
(iv) Communication
(v) Error Detection
(vi) Resource Allocation
(vii) Protection.

Program Execution :-

OS handle many kinds of activities from user programs to system programs like print spooler, name servers, file server etc.

A process includes the complete execution context (code to execute, data to manipulate, registers, OS resources in use).

→ Loads a program into memory.
→ Executes the program
→ Handles program execution.
→ Provides a mechanism for process synchronization and process communication.
→ Provides a mechanism for deadlock handling.

IO Operations :-

An IO subsystem comprises of IO devices and their corresponding driver software.

Drivers hide the peculiarities of specific hardware devices from the users.
An OS manages the communication b/w user and device drivers.
→ IO operation means read (or) write operation with any file (or) any specific device.
→ OS provides the access to the required IO device when required.

File System Manipulation:-

A file represents a collection of related information.
Computers can store files on the disk, for long term storage purpose.
A file system is normally organized into directories for easy navigation and usage.
These directories may contain files and other directories.
→ Program needs to read a file (or) write a file.
→ The OS gives the permission to the program for operation on the file.
→ Permission varies from read-only, read-write, denied etc.
→ OS provides an interface to the user to create and delete files/directories.
→ OS provides an interface to create the backup of the system.

Communication:-

OS manages the communication b/w all the processes in distributed systems.

Multiple processes communicate with one another through communication lines in the network.

The OS handles routing and connection strategies and the problems of contention and security.

→ Both the processes can be on one computer (or) on different computers, but are connected through a computer network.

→ Communication may be implemented by two methods, either by shared memory (or) message passing.

Error handling:-

Errors can occur anytime and anywhere. An error may occur in CPU (or) in I/O devices (or) in the memory hardware.

→ OS constantly checks for possible errors.

→ OS takes an appropriate action to ensure correct and consistent computing.

Resource Management:-

In case of multi-user (or) multi-tasking environment, resources such as main memory, CPU cycles and files storage are to be allocated to each user (or) job.

→ OS manages all kinds of resources using schedulers.

→ CPU scheduling algorithms are used for better utilization of CPU.

Protection:-

Computer system having multiple users and concurrent execution of multiple processes, the various processes must be protected from each other's activities.

Protection refers to a mechanism for a way to control the access of programs, processes, or users to the resources defined by a computer system.

→ The OS ensures that all access to system resources is controlled.

→ The OS ensures that external IO devices are protected from invalid access attempts.

→ The OS provides authentication features for each user by means of passwords.

---

⑦ User Interface and OS Interface:-

The user and OS are connected with each other with the help of interface, so interface is used to connect the user and OS.

In computers there are different interfaces, different types of tasks can be performed by the help of different interfaces.

Command Line Interface:-

The command line interface is an interface whenever the user needs to have different commands regarding the input and output and then a task is performed so this is called

the command line argument. It is used to execute the output and create, delete, print, copy, paste etc. All these operations are performed with the help of the command line interface.

In command line interface multiple commands can be interpreted at same time and executes only one.

The command line interface is necessary because all the basic operations in the computer are performed with the help of the OS and it is responsible for memory management.

Advantages :-

→ Controls OS in application.
→ Faster management.
→ Ability to store scripts
→ Troubleshoot network connection issues

Disadvantages :-

→ Different commands are used in different shells.
→ There present more commands, memorizing is difficult and they complex syntax.

GUI :-

GUI - Graphical User Interface

The GUI is used for playing games, watching videos etc. These are done with the help of GUI because they require graphics.

The GUI is one of the necessary interface because, by using GUI the user can see the picture and play videos.
The basic components of GUI are
→ start menu with programs groups.
→ Taskbar which showing running programs.
→ Desktop screen.
→ Different icons and shortcuts

Choice of Interface :-

The interface that is used with the help of OS for a particular task and that task can be performed with minimum possible time and the output can be shown on the screen In this case we can use the choice of interface

The choice of interface means the OS checks the task and finds out which interface can be suitable for a particular task.

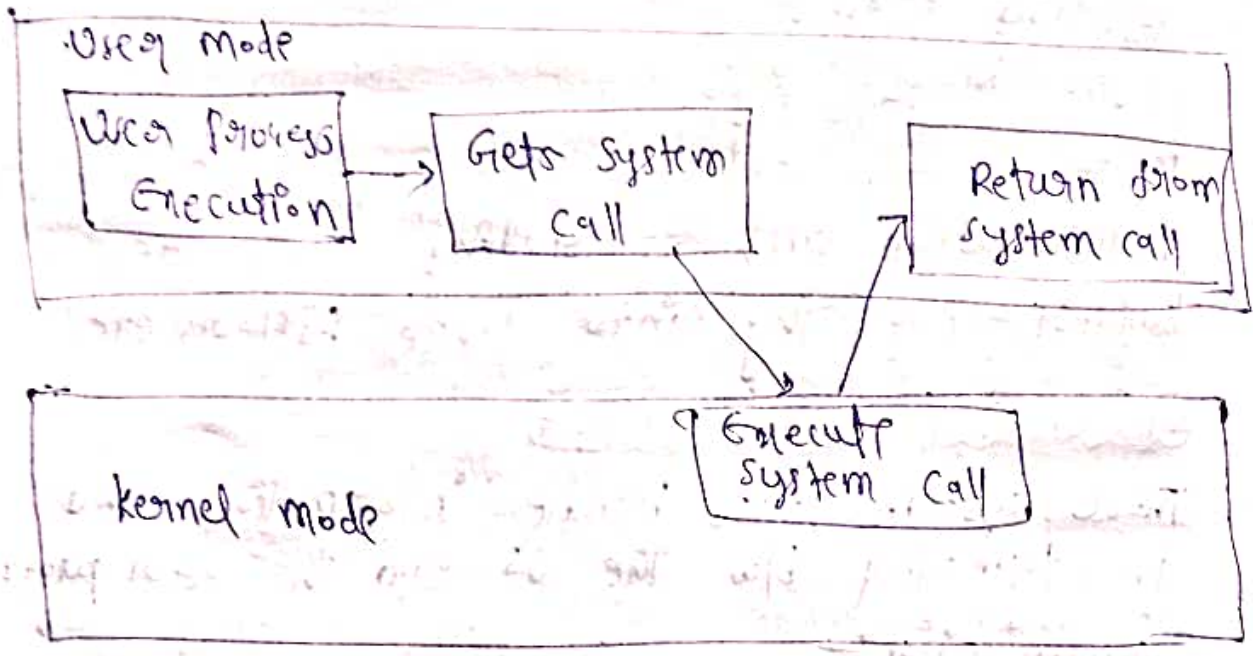
So this type of interface is called the choice of interface and this can be done with the help of an OS.

---

⑧. System Calls :-

The interface b/w a process and an OS is provided by system calls. In general, system calls are available as assembly language instructions.

System calls are usually made when a process in user mode requires access to a resource. Then it requests the kernel to provide the resource via a system call.



The processes execute normally in the user mode until a system call interrupts this. After the execution of the system call, the control returns to the user mode.

→ Creation and management of new processes.
→ Network connections also require system calls.
→ Access to a hardware devices such as a printer, scanner etc requires a system call.
→ Creation/deletion of files and reading and writing from files requires a system call.

Types of System Calls:-

There are mainly 5 types of system calls

Process Control:-

These system calls deals with processes :

such as process creation, process terminati

File Management:-
These system calls are responsible for file manipulation such as creation, reading and writing files.

Device Management:-
These system calls are responsible for device manipulation such as reading from device buffers, writing into device buffers etc.

Information Maintenance:-
These system calls handle information and its transfer b/w the OS and the user progra

Communication:-
These system calls are useful for inter proces communication. They also deal with creating and deleting a communication connection.

Types of System Calls	Windows	Linux
Process Control	CreateProcess() ExitProcess() WaitForSingleObject()	fork() exit() wait()
File Management	CreateFile() ReadFile() WriteFile() ConsoleHandle()	open() read() write() close()
Device Management	SetConsoleMode() ReadConsole() WriteConsole()	ioctl() read() write()

Information Maintenance	GetCurrentProcessID() SetTimer() Sleep()	getpid() alarm() sleep()
Communication	CreatePipe() CreateFileMapping() MapViewOfFile()	pipe() shmget() mmap()

(9) System program:-

In an OS a user is able to use different types of system programs and they are responsible for the development and execution of a program and they can be used by the help of the system calls because system calls define different types of system programs for different tasks.

File Management: These programs create, delete, copy, rename, print, exit and generally manipulate the files and directories.

Status Information: It is the information regarding input, output process, storage and the CPU utilization time how the process will be calculated in how much memory required to perform a task is known by status information.

Programming Language Support: Compiler, assembler, interpreter are programming language support used in the OS for a particular purpose in the computer.

Programming Loading and Execution: To enter the program and after the loading of the progr it needs to execute the o/p of the program and this task is also performed by system by the help of the system programs.

Communication: These services are provided by the user because by using this number of devices communicate with each other by the help of device on wireless and communicati is necessary for the OS.

Background services: There are different types of services are available on the OS for communication and background services are used to chang the background of your window and it also works for scanning and detecting viruses in the computer.

System programs communicate and coordinate the activities and functions of hardware and software of a system and also controls the operations of the hardware.

OS controls the computer hardware and acts like an interface b/w the application software.

Types of System Programs:-

Utility Program:-

It manages, maintains and controls various computer resources. Utility programs are comparitively technical.

The utility programs are antivirus software, backup software and disk tools.

Device drivers:-

It controls the particular device connected to a computer system. These are basically act as a translator b/w the OS and device connected to the system.

Eg:- Printer driver, Scanner driver, storage device driver

Directory Reporting Tools:-

These tools are required in an OS to have some software to facilitate the navigation through the computer system.

Eg:- dir, ls, windows explorer etc.

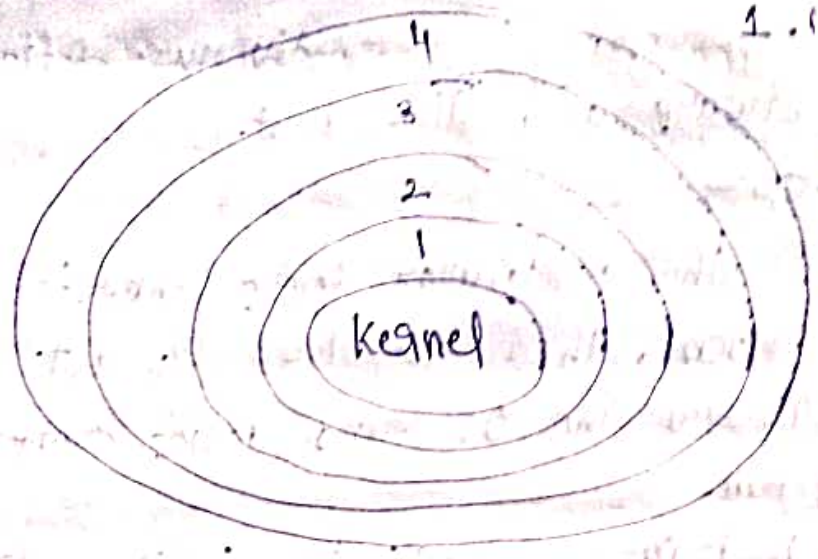
---

(10). OS design and implementation:-

An OS is a construct that allows the user application programs to interact with the system hardware.

OS by itself does not provide any function but it provides an atmosphere in which different applications and programs can do useful work.

There are many problems that can occur while designing and implementing an OS. These are covered in OS design and implementation.



Layered OS design

OS design goals:-

It is quite complicated to define all the goals and specifications of the OS while designing. The design changes depending on the type of the OS.

i.e, Batch system, Time shared system, single user system, multi user system, distributed system etc.

There are basically 2 types of goals while designing an OS.

User goals:-

The OS should be convenient, easy to use, reliable, safe and fast according to the users.

System goals:-

The OS should be easy to design, implement and maintain. These are specifications required by those who create, maintain and operate the OS.

OS Mechanisms and Policies:-

There is no specific way to design an OS as it is a highly creative task.
The difference b/w mechanism and policy is that mechanism shows how to do something and policy shows what to do.
Policies may change over time and this would lead to changes in mechanism.
So, it is better to have a general mechanism that would require few changes even when a policy change occurs.

OS Implementation:-

The OS needs to be implemented after it is designed. Earlier they were written in assembly language but now higher level languages are used.
The first system not written in assembly language was the Master Control Program (MCP).

Advantages of High level languages:-

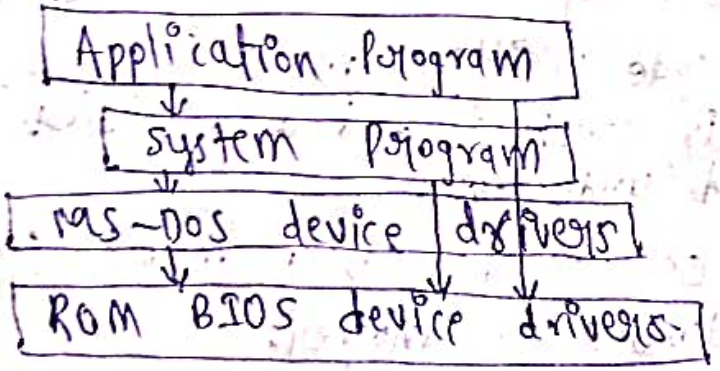
There are multiple advantages to implementing an OS using a high level languages such as
→ The code written more fast.
→ It is compact and also easier to debug and understand.
→ The OS can be easily moved from one hardware to another.

Disadvantages of High level language:-
Using high level languages for implementing an OS leads to a loss in speed and increase in storage requirements.
In modern systems only a small amount of code is needed for high performance, such as the CPU scheduler and memory manager.

(II) OS Structure:-

OS is a complex structure. It should be created with utmost care so it can be used and modified easily. An easy way to do this is to create the OS in parts. Each of these parts should be well defined with clear inputs, outputs and functions.

Simple structure:-
There are many OSs that have a simple structure. These are started as small systems and rapidly expanded more than their scope.
Ex:- MS-DOS.
It was designed simply for a niche amount of people.

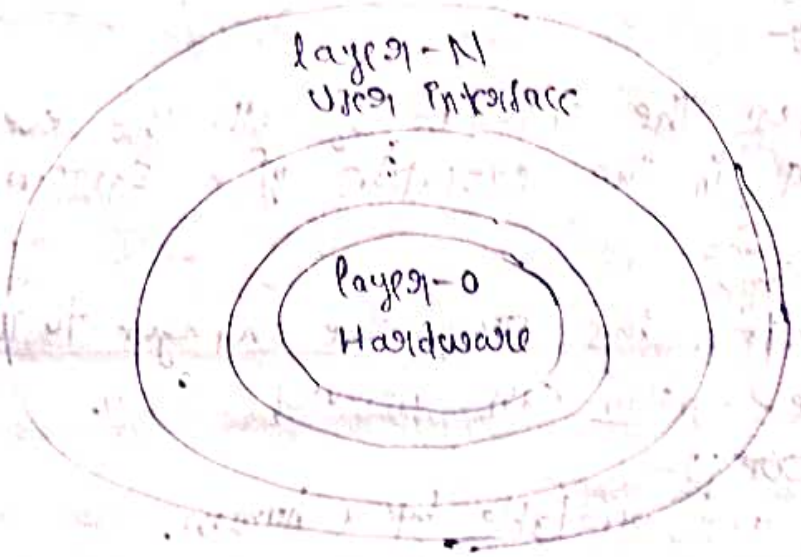


MS-DOS structure

It is better that OS have a modular structure, unlike MS-DOS. The modular structure would allow the programmers to hide information as required and implemented internal routines.

Layered Structure:-

One way to achieve modularity in the OS is the layered approach. In this the bottom layer is the hardware and the topmost layer is the user interface.



From the above image, each upper layer is built on the bottom layer. All the layers hide some structures, operations etc. from their upper layers.

One problem with the layered structure is that each layer needs to be carefully defined.

(12). OS debugging:-

Debugging is the process of finding the problem in a computer system and solving them.
There are many different ways in which OS perform debugging. Some of these are

Log files:-

The log files record all the events that occur in an OS. This is done by writing all the messages into a log file. There are different types of log files

Event logs:-

These stores the records of all the events that occur in the execution of a system.

Transaction Logs:-

The transaction logs store the changes to the data so that the system can recover from crashes and other errors.
These logs are readable by a human.

Message Logs:-

These logs store both the public and private messages b/w the users. There are mostly plain text files and in some cases these may be html files.

Core Dump files:-

The core dump files contain the memory address space of a process that terminates unexpected. The core dump files are used by the developers to find the programs state at the

time of its termination, so that they can find why the termination occured.

- The automatic creation of the core dump files can be disabled by the users. This may be done to improve performance or increase security.

Crash Dump Files:-

In the event of a total system failure, the information about the state of the os is captured in crash dump files. There are 3 types of crash dump files.

Complete Memory Dump:-

The whole contents of the physical memory at the time of the system crash are captured in the complete memory dump.

Kernel Memory Dump:-

Only the kernel mode read and write pages that are present in the main memory at the time of the system crash are stored in the kernel memory dump.

Small Memory Dump:-

Small memory dump contains the list of device drivers, stop code / process and thread information, kernel stack etc.

Trace Listing:-

The trace listing record information about a program execution using logging.

This information is used by programmers for debugging.

System administrators can use the trace listings to find the common problems with software using software monitoring tools.

Profiling:-

It is a type of program analysis, it measures various parameters in a program such as space and time complexity, frequency and duration of function calls, usage of specific instructions etc.

Profiling is done by monitoring the source code of the required system program using a code profiler.

---

(13) System Boot :-

The BIOS, OS and hardware components of a computer system should all be working correctly for it to boot.

System Boot Process:-

The following diagram demonstrates the steps involved in a system boot process.

BIOS
↓
Boot Loader
↓
Kernel
↓
OS Log in.

→ When the power is turned on to start a computer, the CPU initializes itself. This is done by triggering a series of clock ticks that are generated by the system clock.

→ After this, the CPU looks for the system ROM BIOS to obtain the first instruction in the start-up program.

→ The first instruction is stored in the ROM BIOS. and it instructs the system to run POST (Power On Self Test) in a memory address that is predetermined.

→ POST first checks the BIOS chip and then the CMOS-RAM.

→ POST also checks the hardware devices, secondary storage devices such as hard drives, ports etc.

→ After POST, the BIOS finds an OS to load.

→ In most computer systems the OS loads from the C drive on to the hard drive.

→ The CMOS chip typically tells the BIOS where the OS is found.

→ The order of the different drives that CMOS looks at while finding the OS is known as boot sequence.

→ After finding the appropriate boot drive, the BIOS first finds the boot record which tells it to find the beginning of the OS.

→ After the initialization of the OS, the BIOS copies the files into the memory. Then the controls the boot process.

→ Then OS loads the device drivers needed control the peripheral devices.

→ The users can access the system applications to perform various tasks.

Without the system boot process, the computer users would have to download all the software components.

With the system boot, only those software components need to be downloaded that are legitimately required.

This process saves up a lot of space in the memory and consequently saves a lot of time.