

1. Define operating system & what does an operating system do?

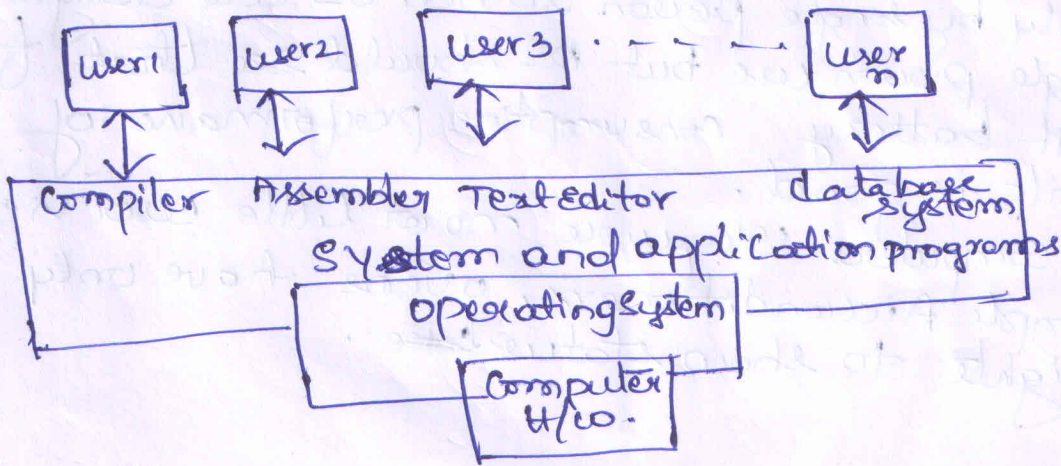
An operating system is a software that manages computer hardware and software resources & provides services for computer programs.

Four components of Computer System are Hardware, operating system, application programs and users.

Hardware means CPU, memory, I/O devices they provide basic computing resources of the system.

Application programs are software such as Microsoft, Excel, compilers, web browsers and other programs written by users to solve their problems.

OS helps us to use resources of system in a proper way.



To understand operating system completely we study it from two viewpoints

- a) users view
- b) system view.

## a) user view:

When only single user is using computer system in that case only attention is paid to have increased system performance, but no attention is paid towards sharing of H/W and S/W resources.

When multiple users are sitting in front of multiple terminals and all are connected to a single system (ex main frame) then all users should share resources of that system. Then OS should increase resource utilization i.e. CPU, memory, other resources should be kept busy most of the time and all users at different terminals should get ~~a share~~ their fair share in ~~not~~ using resources.

In case of hand held devices ~~like~~ <sup>laptops</sup> (mobile phones) ~~and~~ they will be connected either by wired or wireless medium but they will be used only by single person so their OS are designed for single person use but OS should see that for per unit battery consumption performance of mobile is good.

Some computers require more or little user view for example Air conditioners, ovens have only on/off lights to show status ~~like~~.

## b) System view:

Here we can think of operating system as a resource allocator for allocating resources like CPU, memory, I/O devices. OS allocates resources to users and programs. When more than one user uses computer resource allocation is very important.

## Q2. Explain about <sup>②</sup> Computer System Architecture

A Computer System organization can be categorized into depending upon number of processors used in it.

- a) Single processor systems
- b) multi processor systems
- c) clustered systems.

a) Single processor systems: <sup>Computer</sup> Systems with single processor are called single processor systems but they ~~will be~~ can be supported by <sup>disk controller,</sup> Graphics Controllers, I/O processors, etc. But instructions in these programs can be executed only by our CPU. If CPU asks for data from multiple locations on a hard disk, then requests for multiple data items will be stored in queue and will be processed by disk controller but not by our CPU. Like this device specific processors like disk controller, Graphics Controller help our processor or CPU.

b) multi processor systems :- (Also called Parallel Systems or Tightly Coupled systems).  
multiprocessor systems have two or more processors communicating with each other and sharing common bus, clock, memory, peripheral devices.

## Advantages of multiprocessor systems :-

- 1) Increased throughput :- As we have more number of processors in a single system more work is done in less time. (If  $N$  processors are used ~~then~~ speedup ratio is not  $N$  but less than  $N$  because resources should be shared by all  $N$  processors).
- 2) Decreased cost (or) Economy of scale :- multiprocessor systems cost less than multiple single processor systems because they share same memory and I/O devices.
- 3) Increased Reliability :- If ~~function~~ out of many processors even if one processor fails work will not be halted but only it gets slowed down.

Graceful degradation: Ability to continue to provide service proportional to number of processors that are working but not failed is called graceful degradation.

Fault tolerant: Ability to continue to work even if one component ~~is~~ <sup>(CPU or memory)</sup> fails ~~is~~ is called fault tolerant.

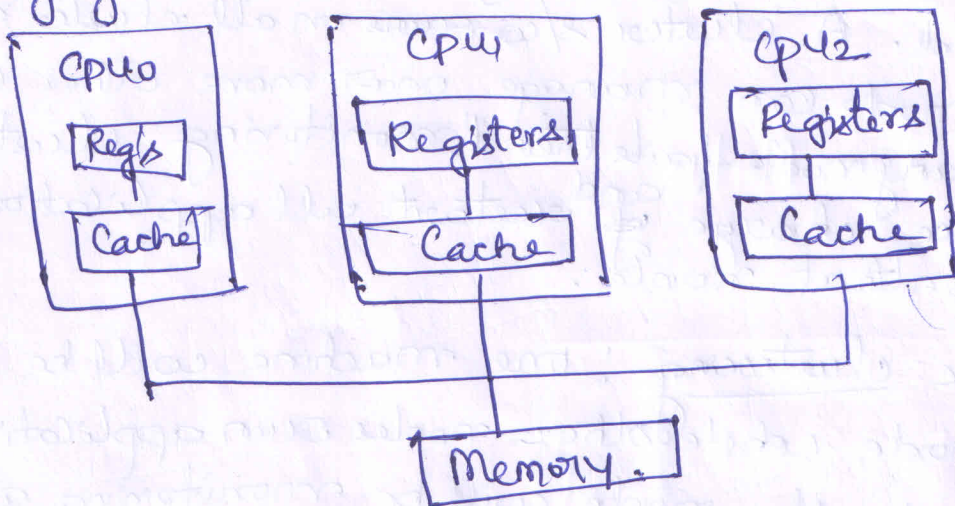
multiprocessor systems are of two types.

1. Asymmetric multiprocessing.
2. Symmetric multiprocessing.

ASMP) Asymmetric multiprocessing :- Here we have master processor and slave processors. Master processor allocates work to slave processor. Slave processor do allocated tasks.

(3)  
(Smp) Symmetric multiprocessing! Here all processors are peers no master slave relationship exists b/w processors. Advantage here is if we have  $N$  processors we can run  $N$  processes on them.

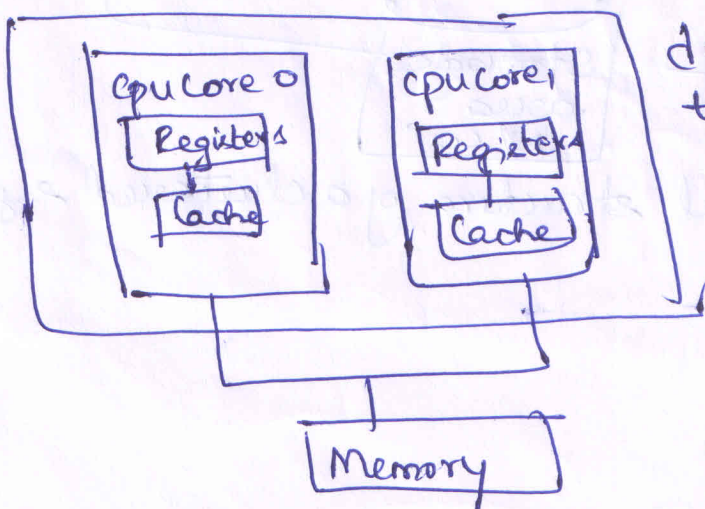
Disadvantage here is  $\therefore$  all  $\text{cpu}$  processors are equal one may get overloaded when other processor is idle.



If we have memory attached to each and every processor then in multiprocessor system we will have multiple memory blocks attached to ~~ind~~ multiple processors.

In UMA (Uniform Memory Access) to access any memory block it takes equal amount of time for any processor.

In NUMA (Nonuniform memory Access) to access a memory block it may take less time for one processor and more time for another processor.



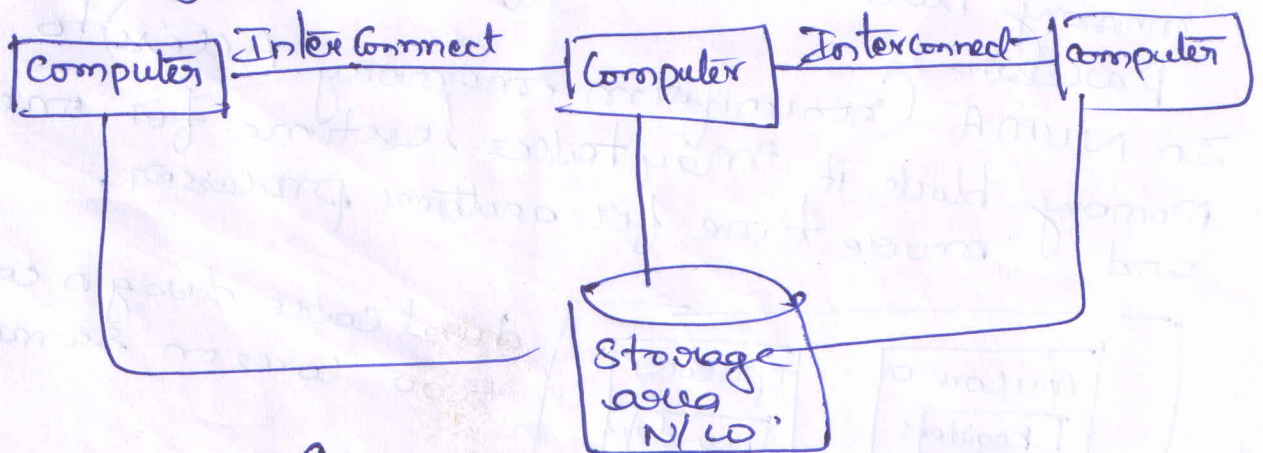
dual core design with two two cores on same chip.

In blade servers multiple processor boards I/O boards and networking boards are placed on same frame or chip.

Clustered systems:- They are composed of two or more <sup>individual</sup> systems that share storage and are closely linked via a LAN (local area network) or some faster interconnection. Clustering provides high availability, i.e. even if one or two systems of cluster fail service will be continued. A cluster s/w runs on all cluster nodes. Each cluster node can manage one or more other cluster nodes. If any node fails then its monitoring cluster node will take charge and restart all applications running on that node.

Asymmetric clustering! one machine will be in hot-standby mode while other nodes run applications. This hot-standby mode will be monitoring active server, if server fails it will become server.

Symmetric clustering!:- Here two or more than two hosts are running applications as well as monitoring each other.



General structure of a clustered system.

Parallel clusters allow multiple hosts to access same data on shared storage

Cluster provides all advantages of distributed system.

clustered system should expand and get all abilities like of a SAN (storage area Network) where many systems are attached to a pool of storage.

Q3 Explain operating system structure (or)

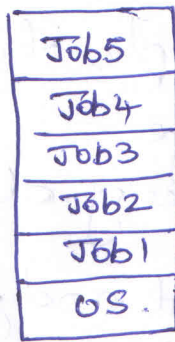
Explain about Multiprogramming & Time sharing systems.

a) Multiprogramming operating system

If we execute single program on computer system then CPU will be idle for most of the time. If a user runs multiple programs on computer then CPU will be made busy & CPU utilization will increase.

Disk contains job pool where set of jobs ready to execute will be kept, from disk some jobs will be brought to main memory & these jobs will be executed in serial order. When job that is executing waits for some I/O operation other job i.e. ready for execution will be assigned CPU. So even if some jobs have to wait for I/O CPU can be busy executing other jobs in the memory.

Using multiprogramming all resources will be utilized efficiently but user cannot interact with computer system.



~~3-6 MVT~~

memory layout for a multiprogramming system.

### Advantages

1. High CPU utilization.
2. It appears as if many programs are allotted CPU simultaneously.

### Disadvantages

1. CPU scheduling is necessary (as many jobs ready for execution or in memory)
2. Since many jobs are stored in memory at same time memory management is needed.

### (b) Time sharing operating system

Time sharing is also called as multitasking and it is similar to multiprogramming. But <sup>in</sup> time sharing user can interact with the system.

In time sharing system CPU executes one user's job for particular ~~#~~ short timeslice and then it executes next user's job for same timeslice. Like this all jobs or user jobs get same amount of CPU time ~~for~~ on round robin fashion. So user gets feel that he is only one using the system. When particular job is getting executed the user of job can give I/P needed ~~to~~ see the O/P on screen.

Program under execution is called as a process.



⑤

Difference b/w Timesharing and multiprogramming systems is in Timesharing CPU executes each program for little amount of time and it automatically switches to next programs in round robin fashion whereas in multiprogramming CPU will not switch from one program to next until first program is completed or it waits for some I/O or other event.

### Disadvantages

1) ~~The~~ Time sharing system takes more response time than multiprogramming systems because ~~cpu~~ in time sharing systems CPU ~~changes~~ <sup>switches</sup> b/w many processes & these processes will be swapped out & in b/w main memory & hard disk.

2) Memory Management is needed

### Advantages

1. Deadlock will not occur as CPU switches b/w different programs.
2. Protection of resource is done.

Q4 Explain about distributed systems (or) distributed operating system

A Distributed system is a collection of physically separate, possibly heterogeneous computer systems that are networked ~~to provide users with~~ <sup>where</sup> can access all the resources that the system maintains.

Access to a shared resource increases computation speed, data availability and reliability.

A network is a communication path between two or more systems. Networks vary by protocols used, distances b/w nodes, transport media. TCP/IP is popularly used protocol supported by almost all operating systems.

Networks are characterized by distances b/w their nodes. LAN (Local area network) connects computers within a room or building. MAN (Metropolitan Area Network) connects computers ~~between~~ within a city. WAN (wide area network) connects computers ~~within~~ b/w countries all over the world.

### Advantages of distributed OS:

1. Resource sharing: S/w libraries, databases, file resources (disk, printers) can be ~~be~~ shared.
2. Higher reliability: Reliability means <sup>high</sup> degree of tolerance against errors & failures.
3. Shorter response time & high throughput.
4. Incremental growth: By adding extra resources to the system functionality of distributed system can be increased.
5. Better price performance ratio: Microprocessor rates are decreased but their <sup>capacity</sup> power is increased ~~to cost~~ ~~to price to cost p~~ ∴ we get good price performance ratio.

## disadvantages of <sup>distributed</sup> OS are (6)

1. Protocol overhead is more than Computation cost.
2. Hard to build well.
3. Successful examples of Distributed systems are scarce.

Q6

Q5 Explain about special purpose systems. (or)  
Explain about Realtime embedded systems, Multi media systems and hand held systems.

Special purpose systems are that class of computer systems a) whose functions are more limited.  
b) whose objective is to deal with limited computation domains.

a) Real Time Embedded Systems :-

Embedded computers are found in Car Engine, ACs, microcoave ovens, Robots etc.  
Using an embedded system we can control heating, lighting, alarm systems and even coffee makers. If we have web access at home, owner can heat up his house before he arrives home.

Embedded systems almost always use real time operating systems. A real time <sup>operating</sup> system is used when rigid time requirements are placed on processor.

Sensors bring data to the Computer. The Computer must analyze the data and adjust controls of corresponding embedded system (like ~~switch~~ <sup>off</sup> or on oven, off A/C etc).

These are two types of real time systems.

Hard real time systems: - It guarantees that critical tasks be completed on time.

Soft real time systems: - They are of less restrictive type. Tasks may be completed little later than the set time.

Here priority scheduling algorithm is used to ~~meet the~~ <sup>assign</sup> priority to tasks.

Examples of Real time Applications are Traffic light system, Air traffic control, Radar systems, Telephone switching system.

Realtime systems require less memory Management than multiprocessor systems.

## b) Multimedia Systems :-

Multimedia data consist of audio, video files as well as conventional files. Multimedia has certain time restrictions. For example to see video without flickering time restriction is 30 frames transmitted in one second.

Multimedia includes audiofiles such as MP3, DVD movies, video conferencing, video clips etc, live webcasts of speeches or sports, webcams that show our shop by sitting at home.

Now multimedia Applications are not only in desktops but also in PDA's and cellular phones.

(c) Hand-held systems: They include PDA's (Personal digital assistants), Palm, pocket PCs and cellular phones ~~using~~ many of which use embedded operating systems.

Handheld systems should have very less size & weight due to which their designing is challenging.

Their memory is also between 1MB & 1GB only. ∴ limited memory of handheld system should be managed efficiently. When task is completed, unneeded memory should be returned to Memory Manager.

Handheld devices need speed processor which consumes more power because of which larger battery is needed. So some handheld systems use low speed processors.

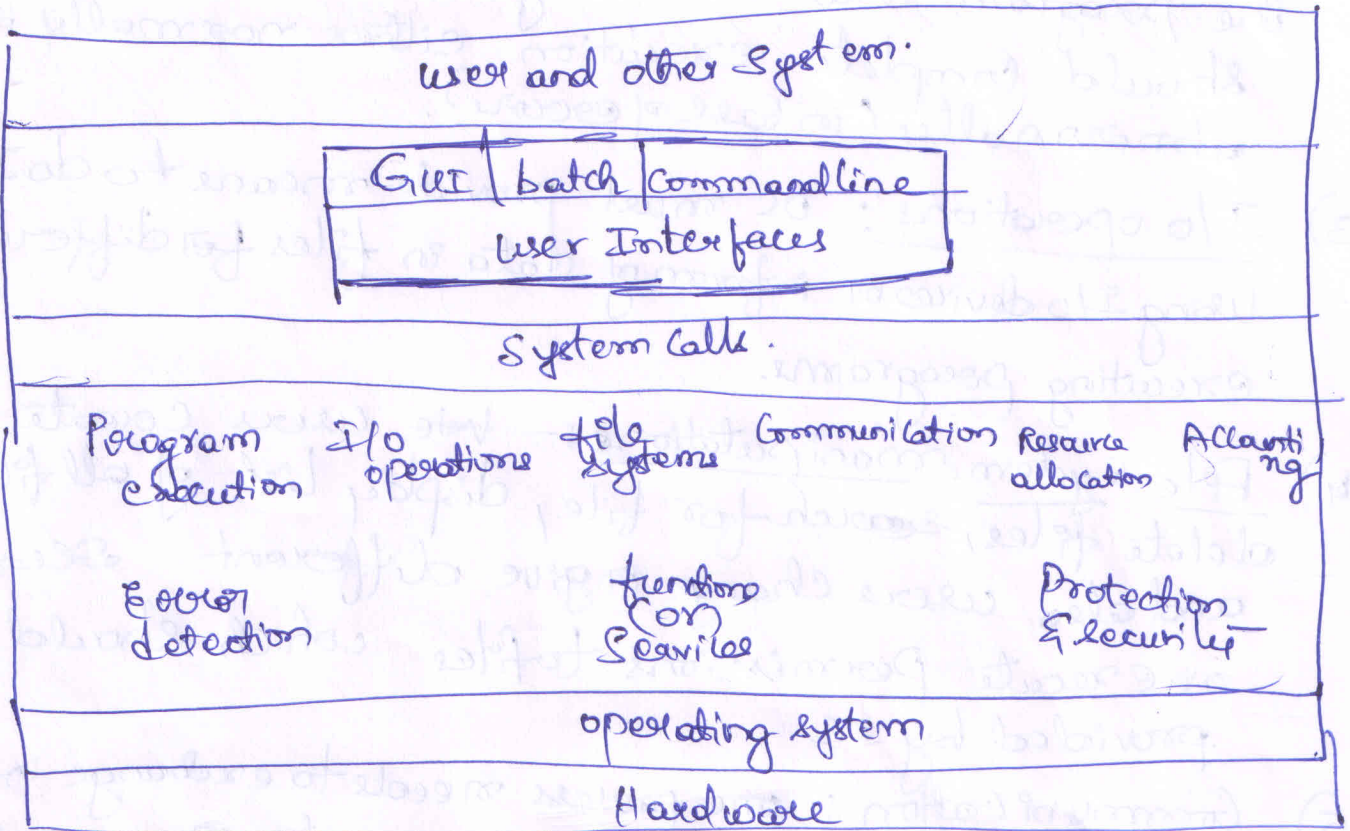
We cannot have I/O devices attached to a handheld device but we can have ~~screen~~ keyboard that appears on 3 inch keyboard of these devices where we can give I/P either by touch or joystick.

Some handheld devices like phones use wireless technology ~~called bluetooth~~ & for ~~remote~~ access to e-mail & web web browsing. Now a days handheld devices are provided with digital cameras, MP3 players

PDA is also called as palmtop, pocket computer, handheld computer. PDA functions as cellular phone, fax sender, web browser and personal organizer.

Q6 Explain Services of operating system (or) Functions of operating system.

OS provides services to programs and to users of those programs.



~~one set of OS services.~~

OS Services or Functions that are helpful to users:-

1. user interface:- We can have 3 types of user interfaces provided by operating system.

Command driven user interface: Here we type or give text commands to OS.

Batch interface: Here all commands and directives to control ~~control~~ commands are put in a file and this file is executed by OS.

Graphical user interface: Here we have icons and a menu we can select option we want by using a pointing device like mouse.

- 2) Program Execution: The OS must be able to load the program into memory and execute it and should complete execution either normally or abnormally (in case of error).
- 3) I/O operations: OS must provide means to do I/O by using I/O devices or inform of data in files for different executing programs.
- 4) File system manipulations: - ~~the~~ users create files, delete files, search for file, display list of all files and also users change or give different read, write or execute permissions to files which should be provided by OS.
- 5) Communication: one process needs to exchange information with other processes on same computer or on another computer by using shared memory or message passing.
- 6) Error detection: - OS should be constantly aware of possible errors  $\rightarrow$  memory (memory error or power failure),  
 $\rightarrow$  I/O devices (parity error, connection failure on NIC, lack of print papers in printers)  
 $\rightarrow$  user programs (arithmetic errors, divide by zero, illegal memory location, too much use of CPU time)

For each type of error OS should take appropriate actions.

OS functions & services for efficient operation of system itself

Resource allocation: When multiple users or jobs are running at the same time resources should be allocated to each of them by OS. Resources means CPU time & I/O devices, Main Memory etc.

Accounting: OS should keep track of which users use how much and what kind of Computer resources. This is Accounting statistical data is used by researchers who wish to reconfigure computer to improve computing services.

Protection & security: When different processes execute concurrently one process should not interfere with other processes or ~~the~~ OS. Also information stored in memory should be protected and should be

Q7 Explain about Protection and Security:-

- When multiple users allow concurrent execution of multiple processes, then access ~~the~~ to various resources should be regulated.
- Resources like file, memory segments, CPU and other resources can be used only by those processes who have ~~got~~ proper authorization of OS.
- A process should execute only within its own address space (i.e. allotted memory) this is ensured by Memory addressing ~~the~~.
- ~~Another~~ Next process can gain control of CPU only if first process ~~relinquish~~ is left by CPU.



- Device control registers that is used to control various peripherals are not accessible to user & thus protected.
- Protection is a mechanism for controlling the usage of resources by different users.
- An unprotected resource cannot defend against user misuse by an unauthorized or incompetent user.
- If a user authentication information like user id, password are stolen their data can be copied or deleted even though file or memory protection is working.
- Attack can spread viruses, worms, identity theft, theft of service (unauthorized use of system) etc, denial of service.
- Prevention of attacks is job of operating system.
- System Security is used to defend a system from internal & external attacks.
- Protection & security system should be able to distinguish b/w authorized & unauthorized users.
- Each user name is mapped to user id and password is assigned to each user id. So if user has to login correct user id & password should be given.
- Generally owner of file has all permissions read, write, execute, delete on a file. If we want to give read permissions to a group of users on that file then instead of giving permissions to them individually we can create a group for that users & give permissions to that group.

Denial of service means an interruption in an authorized users access to computer N/w by which is caused by malicious intent.

(10)

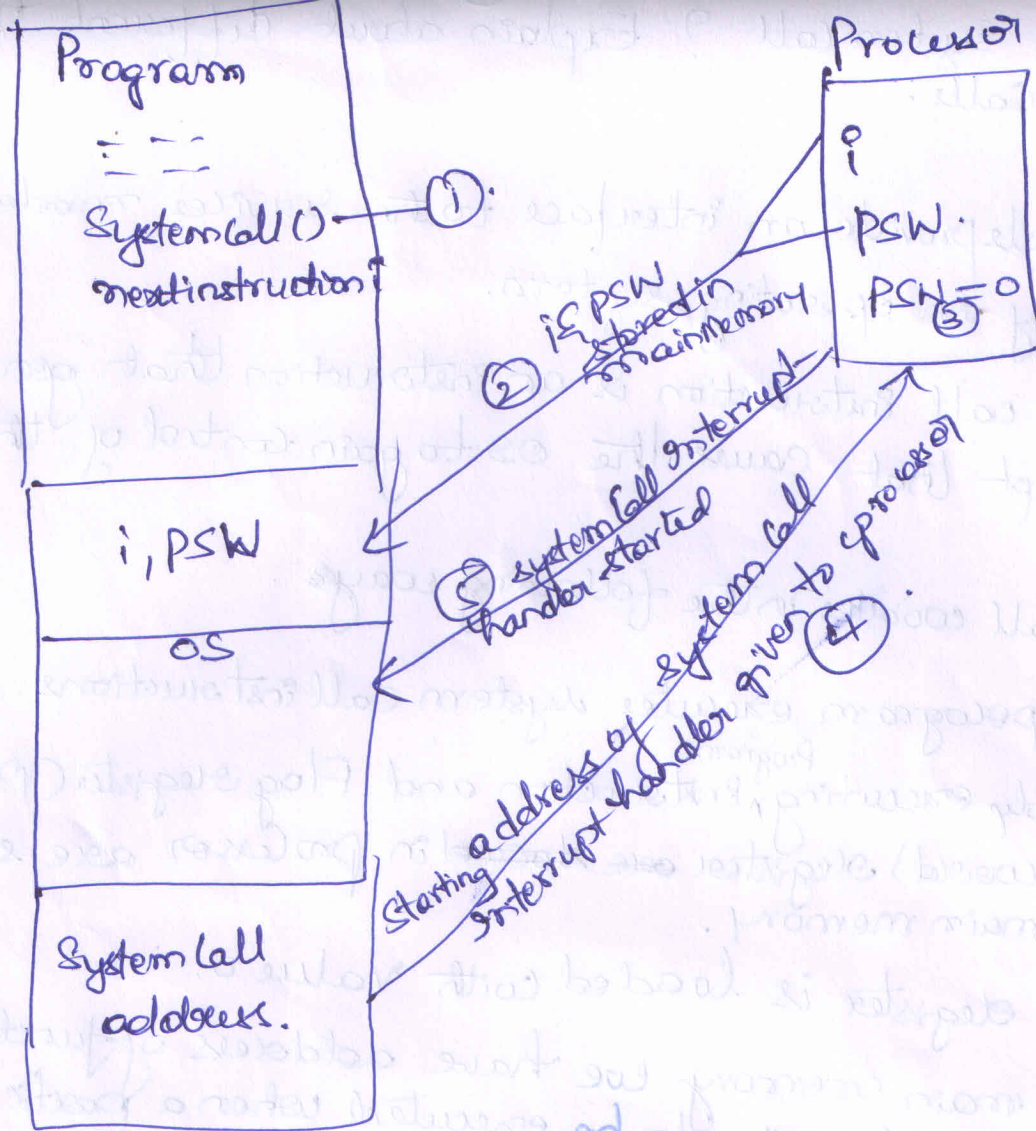
Q. What is a System call? Explain about different types of System Calls.

System Calls provide an interface to the services ~~made~~ provided available by an operating system.

A System call instruction is an instruction that generates an interrupt that cause the OS to gain control of the processor.

System call works in the following ways.

1. Our program executes system call instructions.
2. Currently executing <sup>Programs</sup> instruction and Flag register (PSW program Status word) register ~~are stored~~ in processor are stored into main memory.
3. PSW register is loaded with value 0.
4. In main memory we have address of function (or) interrupt handler to be executed when a particular ~~interrupt~~ <sup>system call</sup> occurs, it will be given to processor.
5. Processor starts executing interrupt handler of that particular system call & completes it:-
6. After completion of interrupt handler of system call first stopped program's next instruction and PSW are again loaded into processor and its execution continues.



Example of how system calls are used. To copy contents of file<sub>1</sub> into file<sub>2</sub> the following system calls will be started.

- Get name of file<sub>1</sub>
- Get name of file<sub>2</sub>
- if file<sub>2</sub> doesn't exist abort.
- Create file<sub>2</sub>, if file<sub>2</sub> already exists with that name abort.
- Read from file<sub>1</sub>, write to file<sub>2</sub> until end of file<sub>1</sub>
- close file<sub>1</sub> & file<sub>2</sub>
- Terminate normally.

## Q8 Types of system calls.

### A) Process control system calls

→ end, abort.

→ load, execute.

→ create process, terminate process.

→ get process attributes, set process attributes

→ wait for time

→ wait event, signal event.

→ allocate & free memory.

→ A running program ~~need~~ halts its execution normally (end system call) (or) abnormally (abort system call).

→ If currently running program is terminated abnormally (or) program has an error and trap is generated then dump of memory is taken & written to disk for examination by debugger.

→ Command Interpreter should get control back to execute next instruction even under normal or abnormal circumstances.

→ In GUI when error occurs appropriate warning message will be displayed, in a command user interface entire process will be terminated.

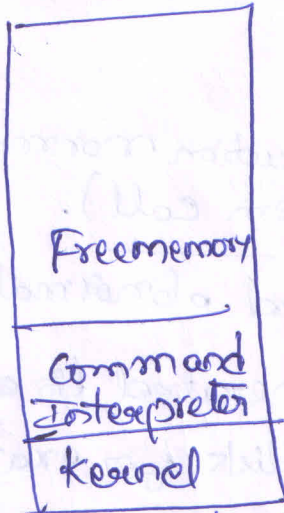
→ Control card is used to manage execution of a process, it defines different error levels. It's like a batch system.

→ load and execute system calls are used to load ~~the~~ ~~the~~ programs into memory & execute them.

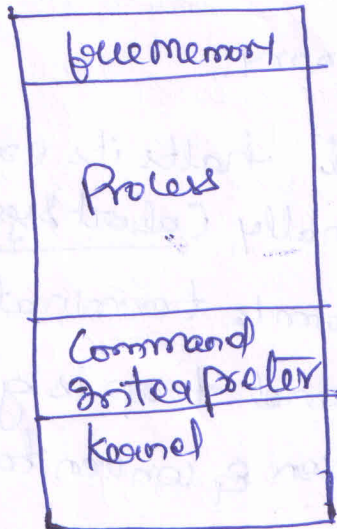
→ Get process attributes and set process attributes are used to know & set process priority, allowed execution time & soon.

→ We terminate a process if it's incorrect or no longer needed.

- If we are waiting for a ~~process~~ certain amount of time to pass (wait time system call), we are waiting for event (wait event system call).
- Process signals when ~~that~~<sup>an</sup> event occurs (signal event).
- Shared data should be provided with acquire & release locks so that only one process can use them at a time.

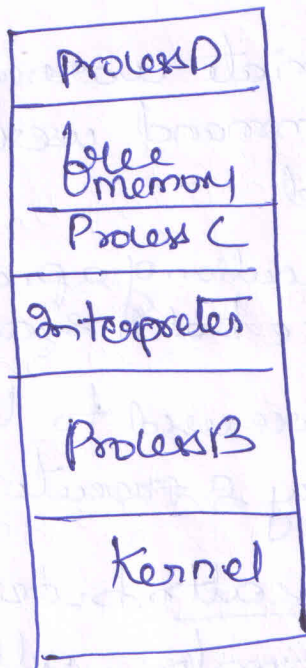


When we start system



When program starts execution

MS-DOS is single tasking system. Program execution is shown in side figure



running multiple programs in UNIX.

Unix is multitasking system

Fork() is used to create child process

exec() is used to load & execute program

## B) File Management System Calls (12)

- Create file, delete file
- open, close
- read, write, reposition.
- get device attributes, set device attributes
- logically attach or detach devices.

First we create file, then open it, write data into file, read data from file, close the file. We may reposition in file either by rewinding or skipping.

File attributes include filename, type, protection codes, Accounting information.

## C) Device management

- Request device, release device
- read, write, reposition
- get device attributes, set device attributes.
- logically attach or detach devices.

→ Resources like Hard disk, Main memory, I/O devices, files are called as devices. Files are virtual devices other are physical devices.

→ When multiple users request ~~use~~ & using resources user first request & gain access to device then release it after use.

→ Memory files, I/O devices can be read, write, repositioned

## D) Information maintenance system calls

Get time or date, set time or date  
get system date, set system date.  
get process, file, or device attribute  
set process, file, or device attributes.

→ These types of system calls are used to transfer information between user & OS.

→ Getting, setting date & time, knowing number users currently using system, version number of OS, amount of free memory etc.

→ These are also useful for debugging programs. Dump of memory for a programme is taken so that if program ends due to some error we can see dump & find error.

→ After execution of each instruction single step Trap int. will be executed by microprocessor, this will be used by debugger.

## E) Communication system calls :-

Create, delete communication connection

send, receive messages

transfer status information

attach or detach remote devices.

Two models for Interprocess Communication are

- (i) Shared Memory model
- (ii) Message passing model.

- Shared memory model is used in multi processor system.
- Message passing is used in multi computer system.
- ~~Messages~~ Before sending or receiving messages connection should be opened b/w sender & receiver
- Each system will have host name & IP address. (Hostid)
- Each process will have process name & process identifier. (PID)
- ~~Source~~ Client is called as source of communication, receiver <sup>of request</sup> is server. Client & servers have daemons.
- Daemons are system programs to create connections b/w sender & receiver.

(F) Protection system calls

In multiprogramming the users resources should be protected from others. For this we can use system calls setpermission and getpermission.  
 If systems are connected in a N/w they should also be protected by unauthorized access through ~~network~~ internet. We can use allowuser & denyuser system calls to ~~to~~ allow or deny users to access resources.



## Q9 Explain about Batch Systems (14)

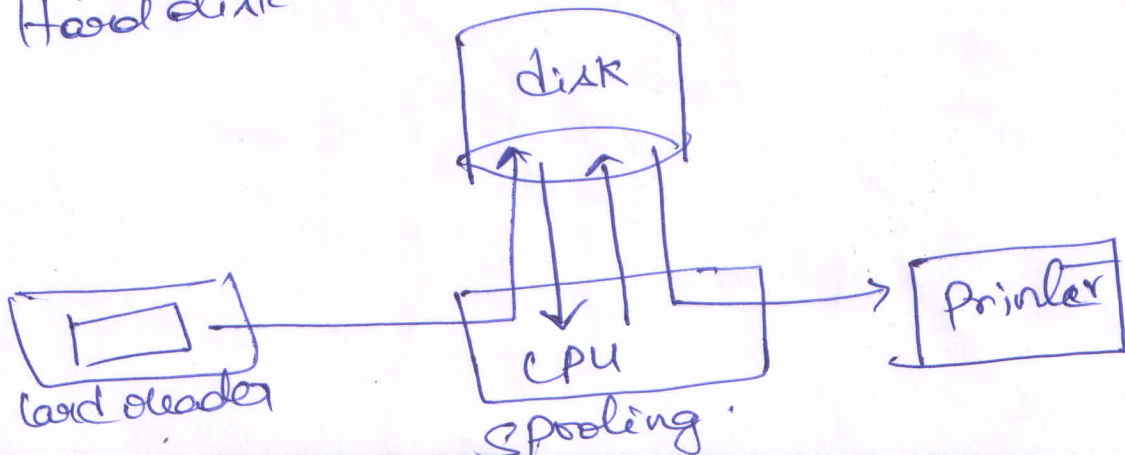
A Batch operating system is one where <sup>Commands, Programs & data</sup> ~~are collected~~ jobs of same type are batched together and submitted to CPU to run as a group.  
Ex: Printing monthly Bank statement of Account holders.

Here after submitting jobs to CPU user cannot interact with system. OP of job will be sent to spool on Hard disk from where it will be sent to printer. ~~They~~ <sup>Batch systems</sup> have no time constraints.

Spooling: Spooling means ~~for~~ simultaneous peripheral operations online. In spooling jobs are put in a buffer, ~~special~~ special area of hard disk from there they can be accessed by various devices like printers etc.

If CPU should give a file to printer for printing CPU's time will get wasted as printer is slower. Instead of that CPU will put file in a spool or buffer in Hard disk from there it will be taken by printer.

Similarly if Card reader wants to give <sup>input</sup> ~~data~~ to CPU, Card reader is slow so Card reader give data to Hard disk which will be taken by CPU.



# Q10 Operating system structure (15)

OS is very large & complex so it should be carefully designed with good structure to function properly. OS will be partitioned into small components rather than being a single monolithic system.

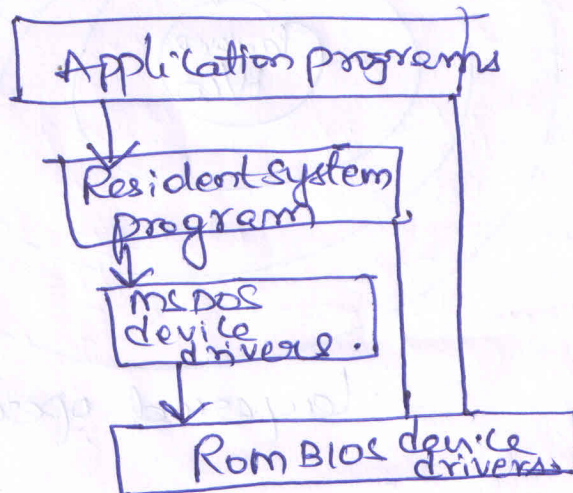
## a) Simple structure

Some OS don't have well defined structure. Some OS started with small limited functions but later they grew beyond scope. MS-DOS is an example of such system.

In MS-DOS different levels of functionality are not well separated. Ex Application programs are able to access basic I/O routines and locate data to devices & memory directly. Because of which MS-DOS is vulnerable to malicious threats & system crashes.

DOS was written for 8088 microprocessor which doesn't provide protection for I/O so DOS leaves I/O accessible to all.

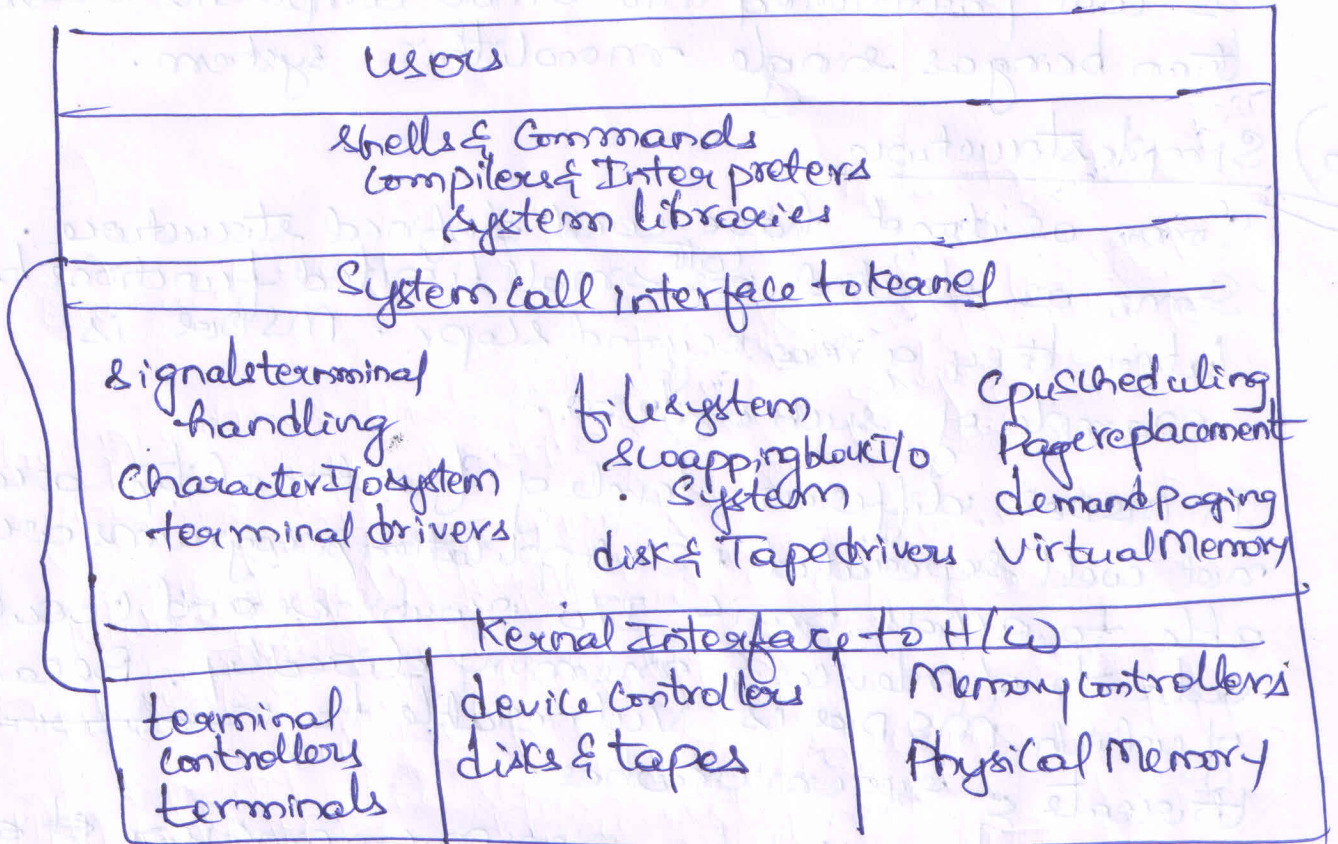
UNIX also started with very few layers as DOS but later it's expanded.



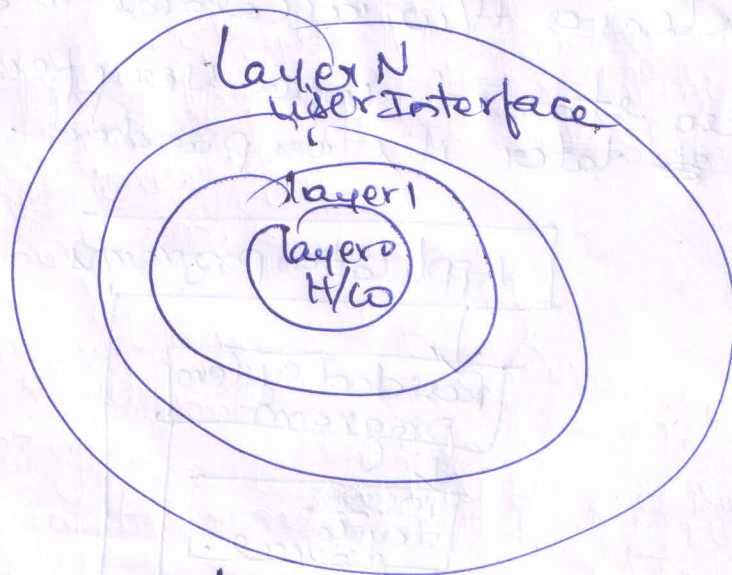
MS-DOS layer structure.

Q what are System Components supported by operating system?

(b) Layered approach



Traditional unix system structure



layered operating systems

OS is broken into number of layers. Layer 0 is H/W and layer N is user interface.

Advantage of layered approach is, 1) simplicity of construction & debugging

- 2) The layers are constructed so that each layer uses only functions of lower level layer.
- 3) If first lower level layer is debugged, then upper level. If error occurs in upper level it then it's only verified as lower level is already debugged.
- 4) Upper level know only functions of lower level layer but not how they're implemented.

Each and every layer contains set of data structures and the routines that will be called by this layer's upper layers.

Here layers should be carefully defined because each layer can use only lower level layer's services.

Disadvantage of layered approach is every request must be passed from outermost Nth layer to layer 0 for execution which wastes time.

## (c) Microkernel

Here Nonessential components are removed from kernel and they are included & implemented at user level.

Micro kernels provide little memory Mgmt and Communication facility.

Advantage of Micro kernel is 1) ease of extending OS.

All new services are added to user space & consequently don't require modification to kernel.

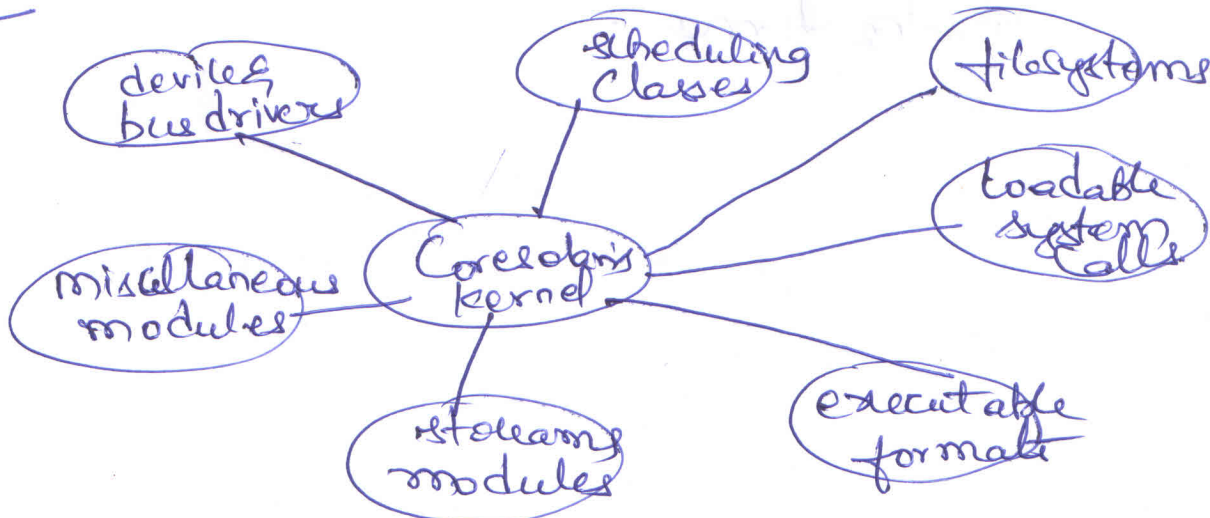
2) OS can be easily ported from one H/W to another as micro kernel is smaller in size.

3) more secure and reliability as most services are running at user than kernel.

Main function of microkernel is to provide communication b/w client program & services provided by kernel. user cannot use services directly rather he interacts with microkernel which provides services to user in form of messages.

Disadvantage of microkernel is their performance decreases due to increased system function overhead.

## (d) Modules :-



For OS design using coprogramming techniques is current approach.

Here kernel has set of core components like device and bus drivers, other components can be linked or loaded when needed at runtime.

Below is OS structure with core kernel and seven types of loadable kernel modules is shown in above figure.

1. Scheduling classes
2. File systems
3. Loadable system calls
4. Executable formats
5. Stream modules.
6. Miscellaneous miscellaneous modules
7. device & bus drivers.

This is more flexible than layered system because any module can call any other module.

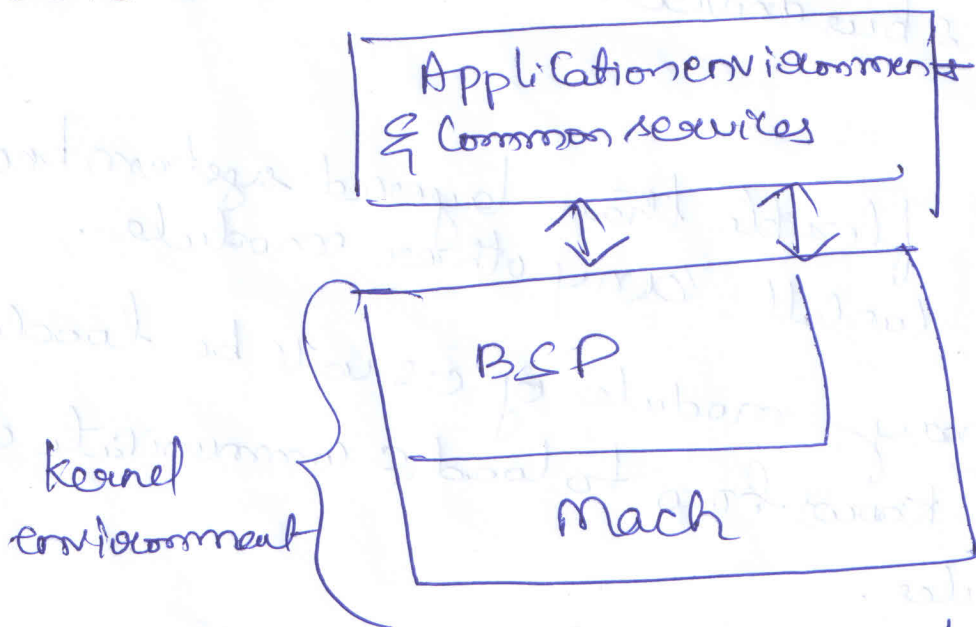
only Primary module of OS will be loaded first & it knows how to load & communicate with other modules.

The Apple macOS X OS uses hybrid structure. one layer consists of Mach microkernel. Top layers include application environments & a set of services providing graphical interface to applications. Bottom layer has Mach micro kernel & BSD kernel.

Mach provides → Memory Management  
 RPC  
 Inter process Communication  
 Message passing  
 Thread scheduling

BSD provides → command line interface  
 → support for Networking & file systems.

kernel also provides with I/O kit for development of device drivers. Applications can use service of Mach or BSD kernel.



MAC OS X Structure.

← I/O over →

## Q11 Explain operating system Generation. <sup>(18)</sup>

An OS should be designed to run on any class of machines with a variety of peripheral configurations. (First H/w on our system, available memory, attached devices, CPU ~~used~~, instruction set used & determined & by system generation program & OS version required for these requirements is then compiled and executable version of OS is generated.)

The system generation program needs the following information about the system for which OS is to be generated.

- what CPU is used?, what instruction set, floating point arithmetic & installed on our system? In case of multiprocessor system each CPU configuration is to be provided.
- How ~~is~~ boot disk formatted? (How many partitions & there on disk.)
- How much memory is available.
- what are devices available? device type & model, address of each device.
- what OS options are desired, how many buffers & needed, what CPU scheduling algorithm to use.



Our OS code is part system out of which we can select needed OS components at run time rather at compile or link time.

After OS is generated we have a program called as bootstrap program or loader which loads OS or kernel into main memory & executes it. ~~It~~  
This process is called Booting

---

Q:2 Explain about Computer System Organization

Ans: Computer system organization are two types

- 1) Computer system operation
- 2) storage structure

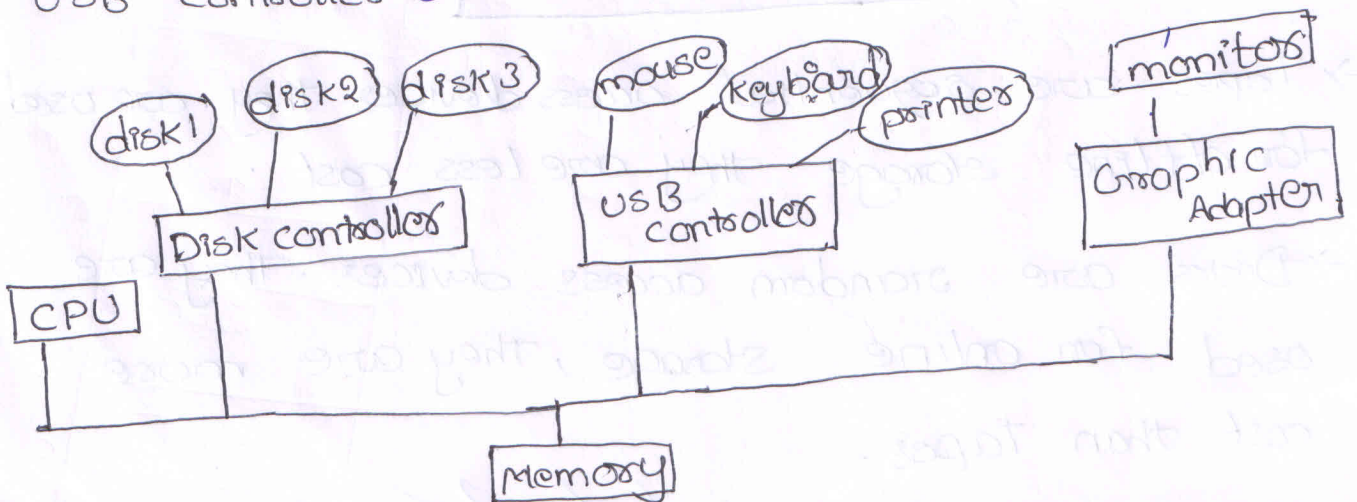
1) Computer System operation

Bootng: Loading of os into memory & executing it is called as bootng.

Boot strap: Boot strap loader is a program which loads os into main memory.

A multi processor has many processors sharing common memory & input/output devices.

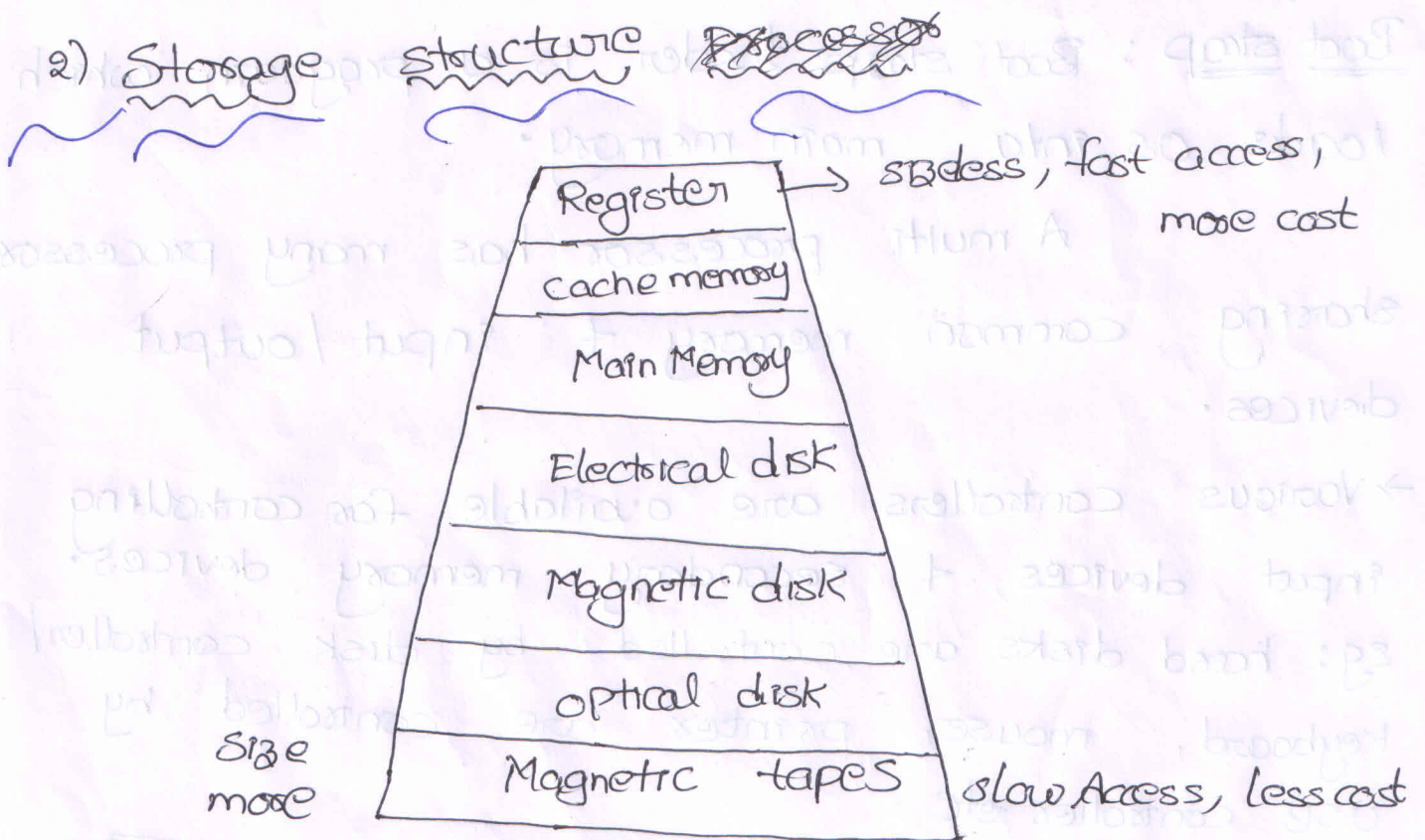
→ Various controllers are available for controlling input devices & secondary memory devices.  
eg: hard disks are controlled by disk controller/  
keyboard, mouse, printer are controlled by USB controller etc.



→ When an interrupt occurs the device which caused the interrupt will send a number using which the address of interrupt service routine will be determined & placed in interrupt vector.

Therefore interrupt vector contains address of ISR (Interrupt service routine).

ISR means it is a routine or function that contains set of instructions to be executed when a particular interrupt occurs.



→ Tapes are sequential access devices they are used for offline storage they are less cost.

→ Disks are random access devices. They are used for online storage, they are more cost than tapes.

✓ ✓ ✓ Q 12 over ✓ ✓ ✓