

Computer Operating System.

[COS]

Manage all the hardware resources of a computer system - **operating system** (Resource Manager)

Managing all the task, incoming and execution what will give what output, what to show on desktop, everything. In short, manages all the task jobs, everything. [Process manager]

All the operations going in computer are looked by OS. interface b/w hardware and instructs the

Taking input from operator and output. to give an output. system to give an output.

Manage inputs. execute inputs. all the

It is also a hardware manager - manages all the hardware resources operating system, their rather their were no software to perform task. were no pre-installed software to execute the task. If you want to run or execute to perform the earlier time you need to perform also.

code for that execution earlier having

Punch card was a card used is passed through certain holes and electricity taken as 1 and 0, if hole was closed, it was 0.

and it is open, it was taken as 1, this is how the data is converted to binary.

That time only hardware knew their and whole execution was to done by people.

But nowadays these execution are done by OS.

Through OS, the speed is increased and system has become multi-tasking. or executing

The ultimate thing from converting, then input to system output that was ~~set~~ hardware, and still it is that only. For ex: if their will be no memory, where system will show output, if their will be no RAM, where code memory will go.

Our operating system is a software and it is hardware dependent. softwares are always installed in some memory.

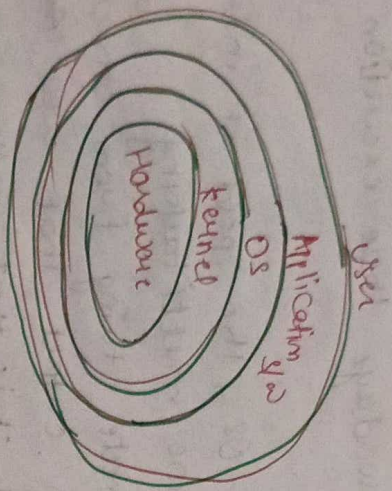
RAM is a volatile memory which have high speed than our hard memory.

Execution is stored in registered memory of RAM and cache.

Cache is a very high speed memory. This cache is not same as volatile

Cache is a pure physical memory. cache seen. Register are faster than chaotic.

OS is installed over hardware, and other applications are installed over operating system.



OS is the layer between hardware and software. It acts as an interface between the user and the hardware. It gives order to the kernel not directly to hardware.

BIOS [Basic Input Output System].

* Examples of well-known OS

1. Mobile OS: Android, iOS, Windows
2. Embedded System OS: - jisme predefined code hai, jab kuch, you cannot change anything. jaise ki car ki engine, aur aise hi.

choke phone me hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

karke hai

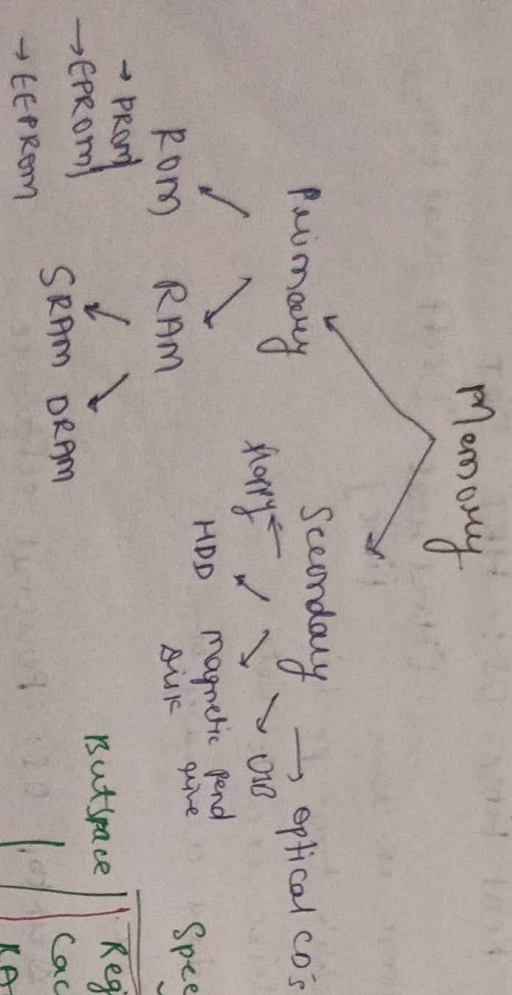
karke hai

karke hai

3. Real Time OS:- HRT, SRT
[Hard real time] [Soft real time]
isme, ek dam hi time pe hi chize execute krni chahiye accuracy of time.
4. Desktop OS: Personal software.
5. Server machine OS:-

* Functions of OS:

- Process Management
high weight process also known as threads
- It creates process, manages process, executes process.
- Memory Management
- Device Management (HDD, printer, monitor, speaker)
- Disk Management
- Network Management (Network card/controller)
- File Management
- Security Management (Logical) through firewall, antivirus, spywares, etc.



* Types of operating system.

1) Batch operating system.

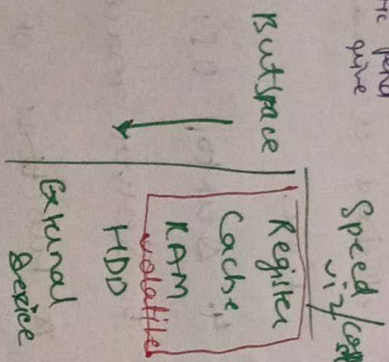
It is a type of OS that allows multiple users to use it at the same time without direct communication b/w them.

2) Multi-programming OS

It can respond to multiple programs at the same time.

3) Multi-Tasking OS

It can also respond to multiple tasks at the same time by context switching.



4) Multiprocessor OS.

OS have more than one processor, but they are working on same physical memory. It will achieve max. throughput.

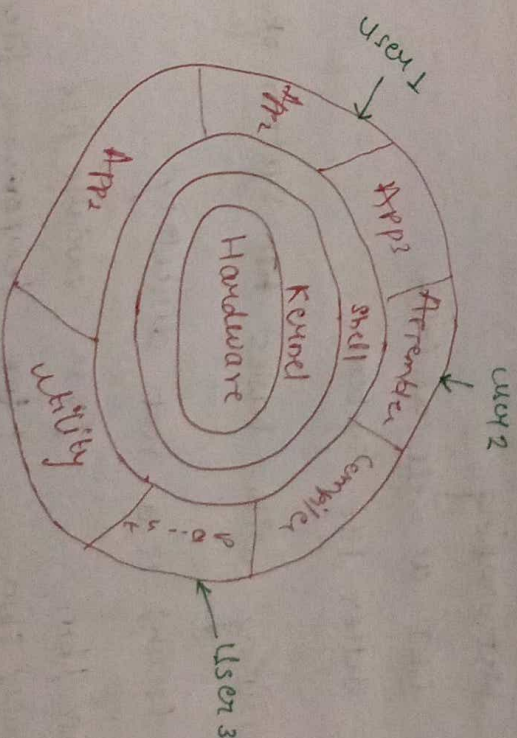
5) Distributed OS.

or processor java hai to dware kahi aur

6) Desk-top OS

7) Server OS.

Linux



Linux is an open source operating system. It is available free to use and users can modify it according to its need.

The founder of Linux is Linus Torvalds. It is available since 1991. An open source community is working behind the update and upgradation of Linux code.

→ Features

1. No cost / Low cost
2. Multi-tasking
3. Security
4. Multi-user
5. Stable and scalable
6. Networking
7. CLI as well as GUI
8. Better file system.

* Working with basic file system of Linux.

/ (slash) is a root directory

User binaries / bin
System binaries /sbin
configuration files / etc
process info. / proc
device files / dev
/var variable file
/tmp temporary files
/usr user programs
/home Root directory of user friendly directory.

* Commands.

1. pwd → present working directory.

nano → editor

Ex: nano abc.txt

If abc file exist it will open that, but if it is not then it will create that.

2. ls → list all the files and directory.

3. touch → creates the file

4. mkdir → makes directory.

5. ls -l → list all files with all details like permission.

* permissions: list

-- -- -- -- --

Read ~
write ~
execute ~

owner group other

6. chmod → to give permissions to file.

Ex: chmod atux file.txt

a = all users
u = owner
g = group
o = other

if - then permission

kafana hai

7. clear → to clear from screen.

8. cd → change directory

9. Cd directoryname / → open desired directory in home

~~rm~~ rm -r directory name → It will delete all the files inside it and also the directory.

rm -r directory name → to delete particular directory.

2. Adding new user

* sudo adduser username

12. To change owner of a file

* chown username filename

* sudo chown username filename

14. To switch user su username

15. To give sudo permissions to a different user
then, etc is a directory

* in root user, go → cd /etc

sudo nano sudoers,
in two lines ALL (ALL) ALL

control X
it will be saved.

* sudo rm -x → A deadly combination

* head -5 filename → It will show 5 lines of file

* head filename → It will show upto 10 lines only if there is 11th line it will not show, it is default

* cat → to display content of file on console

* tail filename → to display bottom lines
tail -5 filename → shows bottom lines 5

* cat > filename → created files and open it in a console.

* Process Management

→ User mode and kernel mode (System call)

* System call types

1. File related calls:- read(), write(), lock(), open(), close(), create(),

2. Process related calls:- New(), fork(), wait(),

Running(),

3. Device Related calls:- Read(), ioctl

4. Information related calls:- getpid(), gettime,

Communication related calls:- wait(), signal(), status().

Process.

It is executable information or code or data which help the processor to execute or complete the user task.

→ units of process

1. code segment
2. Data segment
3. Information segment
4. Memory

① Code segment

It consist of code or data which help the processor to execute.

② Data segment

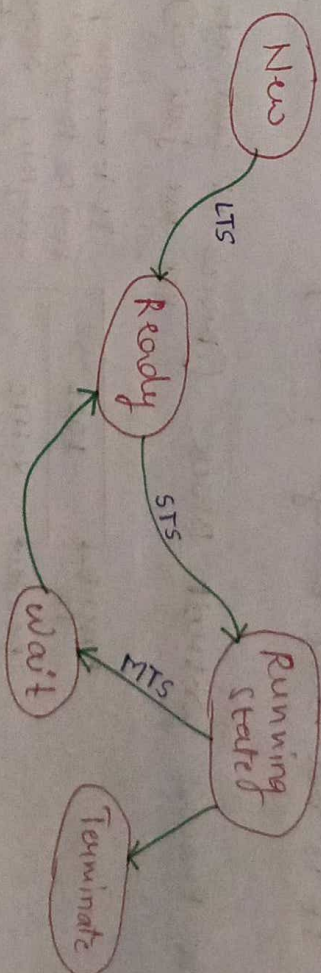
It consist of data required for execution.

③ Information segment

It have metadata about the system available in the process or system which help execution.

★ Process Lifecycle.

1. New state
2. Ready state (Ready Queue)
3. Running state - Processor is executing process
4. Terminated state
5. Wait state



All these are done by process schedulers.

→ Types of process schedulers

1. Long Term Scheduler (LTS)
2. Short TS (STS)
3. Medium TS (MTS)

Starvation

meaning, job or process waiting me hai or ko, duniya jaisi high priority hai, wo run me hai to it will lead starvation.

Occurs in -
- within s
- long on
in non-p
schedulers

Scheduling Algorithms.

Pre-Emptive scheduling

If one process is under execution and CPU uses high priority process in execution, is known as pre-emptive scheduling.

2. Non pre-emptive Scheduling

It is not done through priority

★ Process Control Block. (Process ki saari info process me jati hai)

| | |
|--------------------|--------------------------|
| Parent ID → | 1001 |
| Pointer → | xofffo |
| State → | Ready |
| Allocated Register | A, C, D |
| (Process user) PC | 0xaff0 |
| Allocated Hardware | Keyboard, mouse, printer |
| Accounting info | File.txt, abc.text |
| Priority → | 10 |

PCB P1

It is necessary for OS to understand everything about process, to execute it smoothly.

[Read about all these attributes]

★ Non-Pre-Emptive Scheduling:

① First come First serve.

| PID | Arrival Time | Burst Time | Wait | TAT |
|-----|--------------|------------|------|-----|
| P1 | 0 | 4 | 0 | 4 |
| P2 | 1 | 6 | 3 | 9 |
| P3 | 2 | 8 | 8 | 16 |
| P4 | 3 | 2 | 15 | 17 |

Waiting time = CPU allocation - Arrival time.
Avg. waiting time = Sum of all wait / No. of process

TAT time =

② Short Job First.

Sabse pehle jo process aa jayegi usko execute kar denge par uske execution ke time jo other process queue me usme se dekhenge uska burst time lagya hai, jiska least time sabse kam hoga wo schedule ho jayegi.

| PID | Arrival Time | Burst time | Completion time | Wait | TAT |
|-----|--------------|------------|-----------------|------|-----|
| P1 | 0 | 4 | 4 | 0 | 4 |
| P2 | 1 | 6 | 12 | 5 | 11 |
| P3 | 2 | 8 | 20 | 10 | 1 |
| P4 | 3 | 2 | 6 | 1 | |

TAT = CT - Wait

Round Robin Algorithm

In round robin algo a fixed slice of time is given to each and every process. That slice of time is known as quantum.

4) Priority

* Memory Management.

The memory of computer systems can be divided into blocks of fixed size or variable size. This process is known as fixed size partitioning or variable size partitioning.

Fixed size partitioning:- Here, the memory is divided into fixed size block of size 2 bytes, 4 bytes etc.

- The process which will get the memory may be small or equal to the partitioning block size.
- As the process size may be smaller than block it give rise to internal fragmentation.

Variable size partitioning:- Here the memory is divided into fixed size blocks where all blocks are of sizes like 2 bytes, 4 bytes, 8 bytes, 16 bytes etc.

Internal fragmentation.

Agar process 2 byte ki hai aur memory block 4 byte hai to usko process ke liye extra 2 byte waste hoga.

In variable size partitioning, during contiguous allocation, the size process required in memory is available in it but will not assign memory, this is called external fragmentation.

* Contiguous memory allocation

→ Means memory ke blocks ek ke baad ek ki chahiye iska solution nahi.

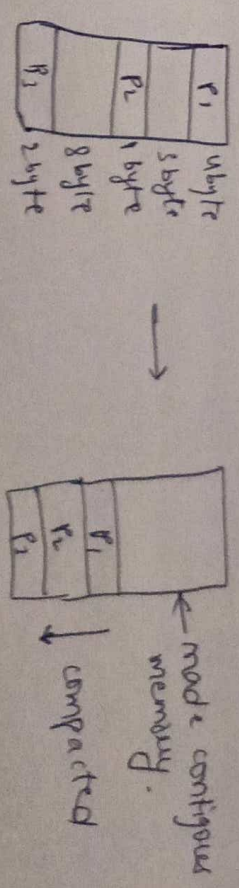
* Non-contiguous memory allocation.

→ Means memory apni apni block me likh sakte hai.

* To solve external fragmentation, we can do

Compaction

Compaction means, memory me jo alag alag block hai usko compact karke ek taraf and then khali space contiguous wali process required ko de do.



* The compaction is a very complex and time taking process.