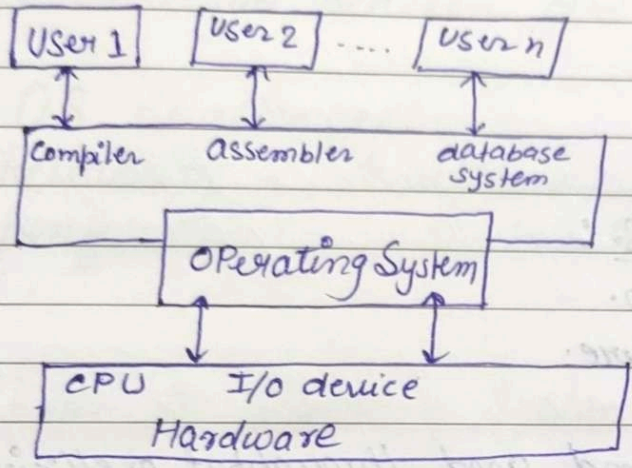
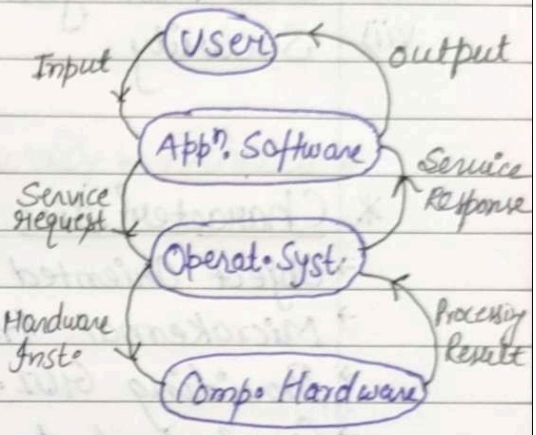
UNIT-1

Dashrath Nandan.

Date : _____

* **Operating System**: An operating system can be defined as an interface between user and hardware. It is responsible for the execution of all processes, managements of CPU, File, etc.





• Static View of System Component

Dynamic View

* **Need of OS:**
  - Convenient to use.
  - Provide abstraction.
  - Easy to use with a GUI.

* **Function of OS:**

i) Memory Management: Keeps the track of primary memory and OS decides which process will get memory when and how much.

ii) Processor Management: Keeps the track of processor and status of process; It is done by traffic Controller.

iii) Device Management: I/O Controller keeps track of all device. Allocates the device in the efficient way.

iv) File Management: File System keeps track of information, location, uses, status, etc. Allocates and de-allocates resources.

v) Storage management

vi) Disk management

vii) Security

* Characteristics of O.S.:
  - → Object Oriented design.
  - → Microkernal Architecture.
  - → Providing GUI.
  - → Convenient to use and good throughput or efficiency.
  - → Increases proformance.

☆ Kernel: The kernal is a computer program at the core of a computer's operating system with complete control over everything in the system.

Features: · Low-level scheduling of process.
· Process Synchronization.
· Inter-process Communication.

Types: (i) Monolithic: It is a single code or block of program.

(ii) Microkernals: Microkernal manages all system resources. These services are implemented in different address space.

Date : _____

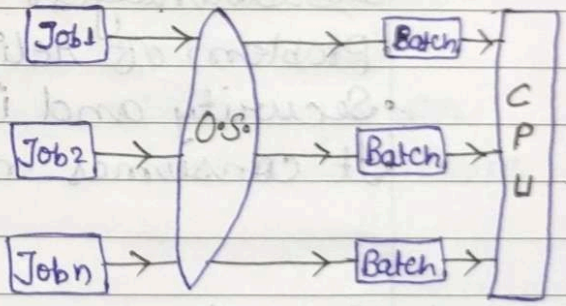
* Application Of OS:
- Security.
- Control over System Performance.
- Job accounting : keeps track of time and Resource.
- Error detecting aids.
- Coordination between other Software and User.

* OS Challenges:-
Reliablity, Availability, Security, Privacy, performance, Portability.

☆ Types Of Operating System.

1) Batch operating System: It is a technique in which operating System collects the program [Punch card, Mag. tape] and data together in a batch before processing starts.

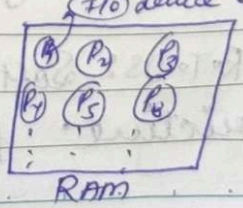
Advantages:	Disadvantages
• Efficient use of resources.	- Limited interactivity.
• Cost-effective	- Delayed processing.
• Reliable	- Time Wastage.
• Easy to use.	- Single task Oriented.



- Non-preemptive.

2) Multiprogrammed OS: Multiprogramming is an extension to batch processing where the CPU is always kept busy.

- It is Non-preemptive.

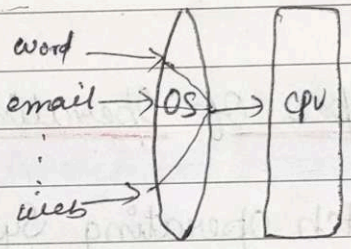


Date :

Advantages	Disadvantages
• Response time can be reduced.	• CPU scheduling is required.
• Hight and efficient CPU utilization.	• It is expensive.

3) Multitasking or Time-Sharing OS:

- It refers to multiple jobs are executed by the CPU simultaneously by switching between them.
- It is preemptive.



Advantages:
- Provide quick response.
- Reduce CPU idle time.

Disadvantages:
- Problem of reliablity.
- Security and integrity.
- It consumes much resources.

4) Real-Time OS: A real time O.S. is the type of system which uses maximum time and resources to output exact and on time result.

(i) Hard R.T. OS: It gurantees that critical task complete on time. For eg. Scientific experiment, weapon system.

(ii) Soft R.T. OS: Soft real time system are less ~~reactive~~ restrictive. for eg. Multimedia, VR, etc.

Date : _____

5) Distributed OS: Distributed system uses multiple central processors to serve multiple real-time applications and multiple user.

- Not Centralized.
- Loosely Coupled System.

Advantages:
- Speedup the exchange data with one another.
- Better Service to customer.

Disadvantages:
- Complex System.
- Security problem.

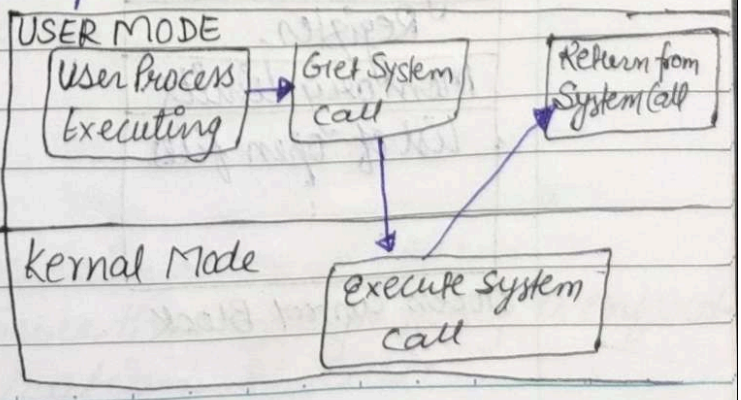
6) Network OS: This System runs on a Server and provide the server the capability to manage data, user, groups, Security, and other networking function. (LAN or private network)

Advantages
- Centralized Server are highly Stable.
- Remote access is possible.

Disadvantages:
- High cost.
- Dependency on central location.

☆ System Calls:
A System call is a mechanism that provides the interface between a process and the OS.

⇒ It is a programmatic method in which a computer program request a Service from the Kernal of OS.



Date :

Types :

System Call
- → File Management : Open(), Read(), Write(), Close, Create file, etc.
- → Device Mangement : Request & release device,
- → Information Maintenance : get Pid, get time & Date.
- → Process Control : load, Execute, abort, fork, wait
- → Communication : Pipe(), Create/delete file

⇒ System calls provide interface between the process and OS.

System programs provide basic functioning to user so that they do not need to write their own environment for program development and execution.

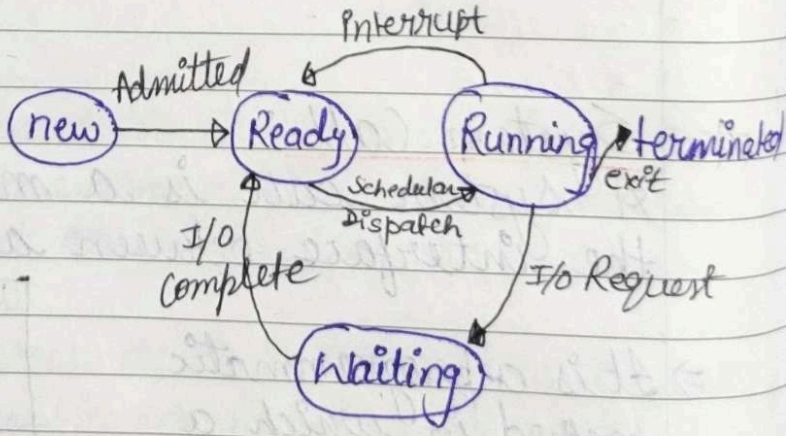
☆ Process Management :

A program in execution state is called a process.

Process State :

Process State
Process Number
Program Counter
Register
Memory limits
List of open files
:

Process Control Block



Process State

As a process executes, it changes states.

Date :

A process Control Block (PCB) Contains information about the process, i.e. registers, quantum, priority, etc. The Process table is an array of PCB that Contains- Pointers, Process state, Process Number, Program Counter, etc.

* Remote Procedure Call :

A remote procedure call is an inter-process Communication technique that is used for client-Server based application. It also known as Subroutine or function calls.

- Advantage of RPC :
  RPC support process oriented and thread oriented models.
  Abstraction of internal messaging mechanism.

- Disadvantage of RPC :
  There is no flexibility in RPC for hardware architecture.
  Expensive because of RPC.

☆ CPU Scheduling :

CPU Scheduling is a process that allows one process to use the CPU while another process is delayed due to unavailability of resources such as I/O etc, to make full use of CPU.

* Objectives of Scheduling :

- Fairness : Each Process gets its fair share of CPU.
- Policy Enforcement : Make Sure that System's policy is enforced.
- Efficency : Full optimization of CPU.

Date :

- Response Time : Schedular should minize Response time.
- Turnaround : Schedular minimize waiting time for output
- Throughput : Schedular should Maximize the number of jobs per unit time.

* Preemptive Scheduling	Non-Preemptive
• The resource are assigned to a process for long period of time.	- Once resources are assigned, they are held until it completes.
• Its process may be paused in the middle of execution.	- Once a process is started, it must be completed, it not be interrupted in middle.
• It is flexible.	- It is rigid.
• Its CPU utilization is very high.	- Its CPU utilization is very low.

* CPU Scheduling Algorithm : A process schedulers schedules different process to be assigned to the CPU based on particular Scheduling Algorithm.

CPU Scheduling Algo.

Preemptive	Non-preemptive
→ Shortest Remaining Time (SRT)	→ First Come First Serve (FCFS)
→ Round Robin Sched. (RR)	→ Shortest Job Next (SJN)
→ Priority Based Sched.	→ Priority Based Scheduling ( )

Date :

☆ Criteria Used for CPU Scheduling Algorithm :

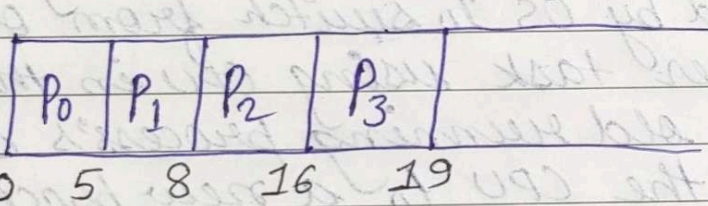
CPU Utilization, Throughput, Arrival Time, Burst Time, Turnaround Time, Waiting Time, Response Time, Exit Time.

☆ FCFS : First Come First Serve

(execution Time) TAT = CT - AT (TAT - BT)

P.NO	AT	Burst Time	Completion Time	Turnaround Time	Waiting Time
P0	0	5	5	5	0
P1	1	3	8	7	4
P2	2	8	16	14	6
P3	3	6	19	16	10

Grantt chart



: Total execution time = 19 units

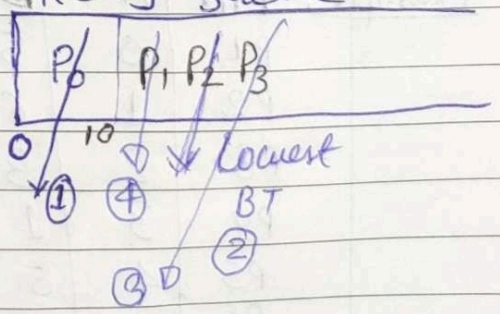
☆ SJN (Shortest-Job-Next Scheduling) :

or Shortest Job first : Without preemption

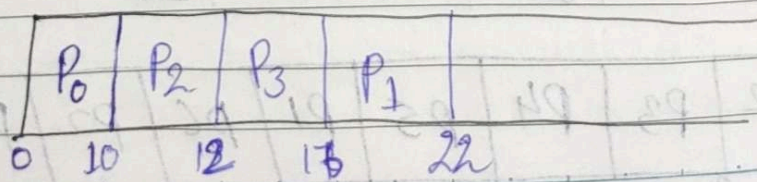
Criteria :- Burst Time

P.NO	AT	BT	CT	TAT	WT
P0	0	10	10	10	0
P1	1	6	22	21	15
P2	3	2	12	9	7
P3	5	4	16	11	7

Ready Queue



Gantt Chart



TAT = CT - AT
WT = TAT - B.T.
RT = CPU First Time - A.T.

Date :

★ SRT (Shortest Remaining Time):

SRT is a preemptive version of SJN algorithm.

★ Priority Based Scheduling:

- It is a non-preemptive algorithm OR Preemptive also.
- Each process is assigned a priority. Process with Higher priority is to be executed first and so on.
- Process with same priority are executed on First come first served basis.

★ Round Robin Scheduling:

Context Switching :- The context switching is a technique used by OS to switch from one task to another task using CPU in the System. It stores the old running process's status and assign the CPU to a new process to execute its task.

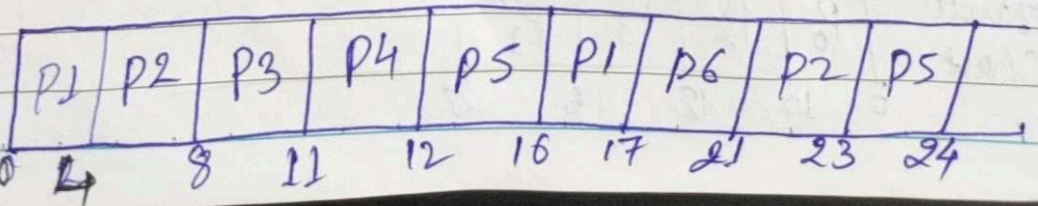
⇒ RR is a preemptive scheduling algorithm.
⇒ Criteria :- Time quantum.

Example:

Pid	A.T.	B.T	C.T	TAT	WT	RT
P1	0	5	17	17	12	0
P2	1	6	23	22	16	3
P3	2	3	11	9	6	6
P4	3	1	12	9	8	8
P5	4	5	24	20	15	8
P6	6	4	21	15	11	11

Time Quantum = 4 sec (Given)

Gantt Chart



Date :

Ready ques (0) ⇒ RQ at 4 Sec → BT=(5-4=1)

P1 | P2 P3 P4 P5 P1

0 4

execute for next 4 sec

Context Switching

⇒ RQ at 8 Sec:

P6 P4 P5 P1(1) P6 P2(6-4=2)

terminates after 3 sec.

⇒ RQ at 11 Sec.

P4 P5 P(1) P6 P2(2)

1 Sec then terminate

⇒ RQ at 12 Sec.

P5 P1(1) P6(4) P2(2)

4 Sec execution

⇒ RQ at 16 sec.

P1(1) P6 P2(2) P5(5-4=1)

1 sec than terminate

⇒ RQ at 17:

P6 P2(2) P5(1)

⇒ RQ at 21 → last terminate

P2(2) P5(1)

2 sec than terminate.

Date :

☆ Process Synchronization:

On the basis of Synchronization, process are of two types :-

(i) Cooperative process is one that can affect or affected by other processes executing in the system. Share common resources like code, CPU, etc.

(ii) Independent process: Execution of one process does not affect the execution of other process.

- When multiprocess run on a system, then there is two mode :-
  (i) Serial execution (ii) Parallel execution.

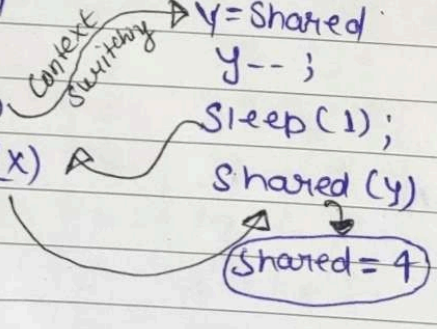
Process Synchronization is the way by which processes that share the same memory space are managed in an Operating System.

☆ Need of Synchronization

(1) Race Condition:

Shared = 5

Thread 1	Thread 2
X = shared	
X++ ;	
sleep(1);	Y = Shared
	Y-- ;
	Sleep(1);
Shared (X)	Shared (Y)



This is wrong, since 1 is added and 1 is subtracted from 5, so, the answer should be 5, but it giving 4 or 6. This problem is known as Race Condition.

(2) Critical Section problem: $\{P_0, P_1, \ldots P_{n-1}\}$, Each process has a segment of code, called a Critical Section, in which the process may be changing the common variable, etc.

* Mutex lock. youtube/wordfile
* Semaphore.

Date :

do{	The critical section problem is to design a
entry section	protocol that the processes can use to
Critical Section	cooperate.
exit section	In this, when one process is executing in
remainder	its critical section, no other process is
} While (true);	allowed to execute in its critical section.

* Three requirements for critical problem solution :-

(i) Mutual exclusion: If a process P_i is executing in its critical section then no other process can be executing in their critical section.

(ii) Progress : If no process is executing in Critical Section, then only those process that are not executing in the remainder section can participate.

(iii) Bounded Waiting: There exist a bound on the number of times that other process are allowed to enter critical section.

(3) Peterson's Solution: A classic Software based solution to the Critical-Section Problem.

int turn indicates whose turn it is to enter its Critical section.

Bool flag indicated if a process is ready to enter C.S.

Structer of P_i

```
do{
    flag[i] = true;
    turn = j;
    while (flag[j] && turn == [j];
        Critical section
    flag[i] = false;
        remainder section
} while (true);
```

Structure of P_j

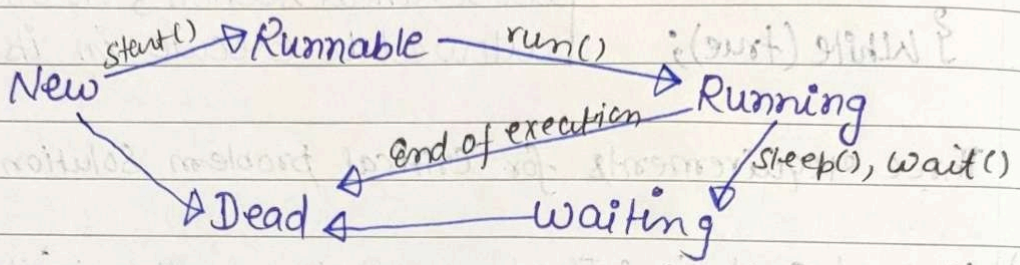
```
do{
    flag[j] = true;
    turn = i;
    while (flag[i] && turn == [i];
        Critical section
    flag[j] = false;
        remainder section
} while (True)
```

Date :

☆ Threads: Types → in OS → Kernel level Thread
User level Thread.

A thread is a single sequence stream within a process. They are also called lightweight processes.

Life Cycle of Thread :-



* User-level Threads	Kernel level Threads
(i) Thredd management is done by application.	Thread management is done by the Kernel.
(ii) Faster to create & manage.	Slower to create & manage.
(iii) It is generic and run on any OS.	It is specific to the OS.
(iv) Multi-threaded applicn. cannot take advant. of multiprocessing.	Kernel routines themselves can be multithreaded.
(v) POSIX Pthread, Mach C-thread.	Windows NT, Windows 2000.

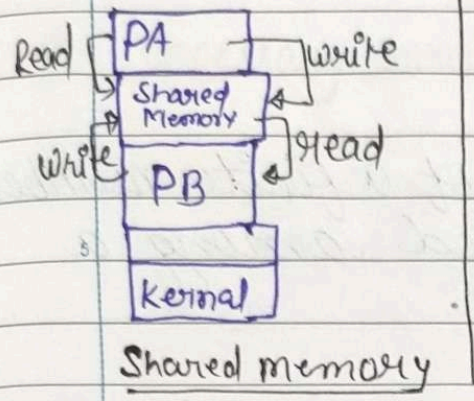
☆ Inter process Communication :-

Cooperative process want to communicate with each other and want to exchange information or data, then we have two models :-

(1) Shared Memory (2) Message Passing

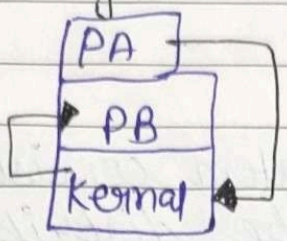
(1) Shared memory :- Particular section of memory is shared among the processes for communication purpose.

Date :

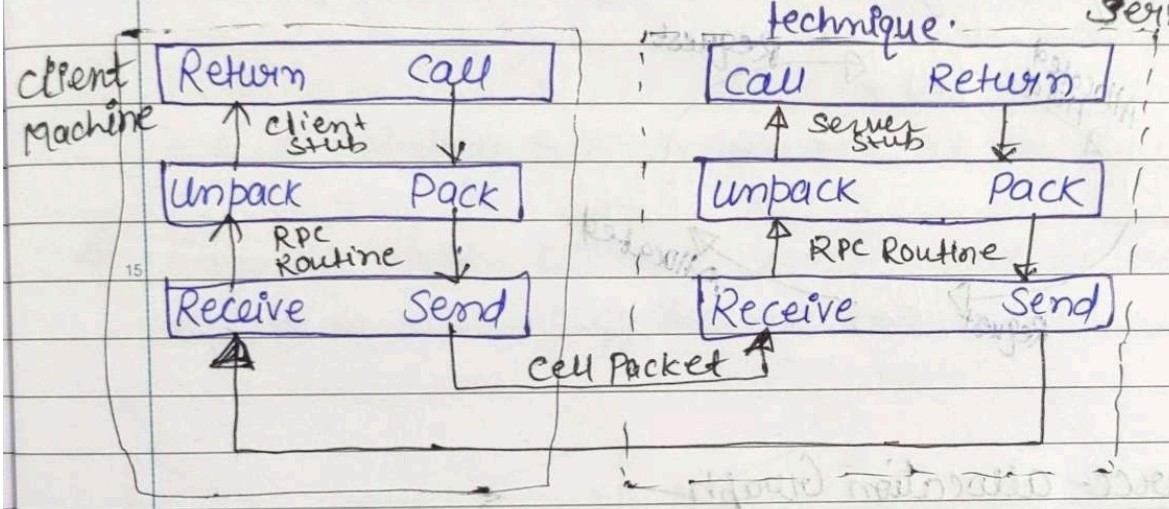


Shared memory

(2) Message Passing.



* Remote Procedure call :- It is an inter-process communication technique.



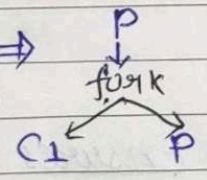
☆ Fork() System call : The fork system call is used to create a new process.

fork → 0 child
→ +ve Parent
→ -ve child X

Eg

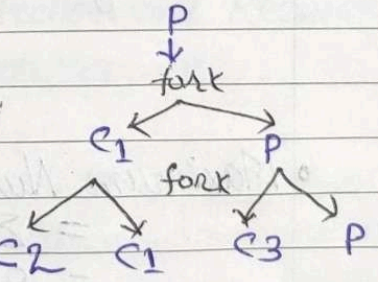
```
main()
{
fork()
Printf(Hello)
}
```

⇒



Eg

```
main()
{
fork()
fork()
Printf (Hello)
}
```



Unit: 1.3

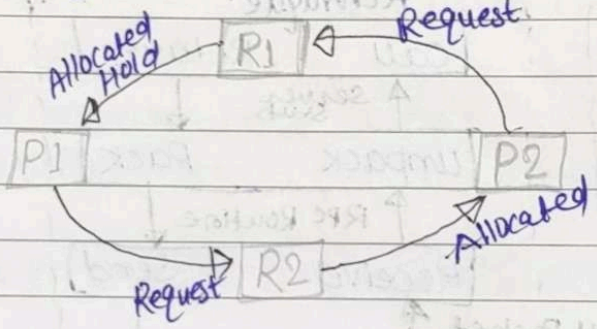
Date :

Dashrath Nandan

☆ Deadlock :-

System Model: A system consists of a finite number of resources to be distributed among a number of competing processes.

* Deadlock is a situation where each of the computer process waits for a resource which is being assigned to some another process.



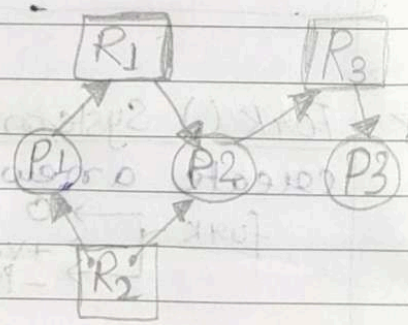
* Resource-allocation Graph

Process :- ○

Resource :- [: :]

Pi request for Resource :- (Pi) → [: :] R

Pi holding resource :- (Pi) ← [: :] R



- Maximum Number of Units that Ensures Deadlock
  $= \Sigma x_i - n$
  = Sum of max needs of all n process − n.

- Minimum Number of unit that Ensures No Deadlock
  $= (\Sigma x_i - n) + 1$

Date :

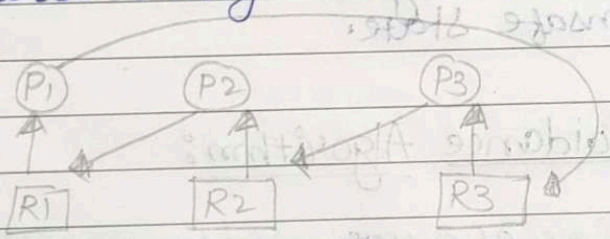
* Necessary Conditions for Deadlock:

1) Mutual Exclusion: At least one resource must be held in a non-shareable mode. It implies, two processes cannot use the same resource at the same time.

2) NO Preemption: Resources cannot be preempted, i.e. a resource can be released only by the process holding it, after the process has completed its tasks.

3) Hold and Wait: A process waits for some resource while holding another resource at the same time.

4) Circular Wait: All the process must be waiting for the resource in cyclic manner.



* Deadlock Handling Strategies
  - Deadlock Prevention
  - Deadlock Avoidance
  - Deadlock Detection and Recovery
  - Deadlock Ignorance

(1) Deadlock Prevention: For a deadlock to occur, each of 4 necessary condition must hold. By ensuring that at least one of these conditions cannot hold, we can prevent the occurrence of a deadlock.

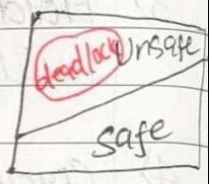
(1) Mutual Exclusion (3) Hold and Wait
(2) NO preemption (4) Circular Wait.

Date :

2.) Deadlock Avoidance :- This strategies involves maintaining a set of data using which a decision is made whether to accept new request or not. If accepting new request cause the system to move in a unsafe state, then it is discarded.

Safe state : A state is safe if the system can allocate resources to each process (upto its Maximum) in some order and still avoid a deadlock.

If System is in Safe State ⇒ No deadlock
If System is in Unsafe State ⇒ Possiblity of deadlock



Avoidance ⇒ Ensures system will never enter an unsafe state.

* Deadlock Avoidance Algorithm:

- Single instance of a resource type:
  → Use a resource-allocation graph

- Multiple instance of a resource type:
  → Use the banker's algorithm.

☆ Banker's Algorithm :-
Banker's algorithm is a deadlock avoidance strategy. It is so called because it is used in banking system to decide whether a loan be granted or not.

Date :

Data structure (Banker's Algorithm)

Available · Max · Allocation · Need

☆ Algorithm :

Process (Max, Need, Allocation) ← any 2 given

m = no. of resources
n = no. of processes

Step 1: Flag[i] = 0 for i = 0, 1, ..., (n-1) & find Need[n][m] = Max[n][m] − allocation[n][m]

Step 2: find a Process P_i such that :-
Flag[i] = 0 & Need$_i$ <= Available

Step 3: If such i exits then-
Flag[i] = 1, available = available + Allocation$_i$

Else,
goto Step 2.

Ques: Total ⇒ A = 10, B = 5, C = 7

⇒ Max Need − Allocation

Process	Allocation			Max Need			Available			Remaing Need		
	A	B	C	A	B	C	A	B	C	A	B	C
P1	0	1	0	7	5	3	3	3	2	~~7~~	~~4~~	~~3~~ ④
P2	2	0	0	3	2	2	5	3	2	~~1~~	~~2~~	~~2~~ ①
P3	3	0	2	9	0	2	7	4	3	~~6~~	~~0~~	~~0~~ ⑤
P4	2	1	1	4	2	2	7	4	5	~~2~~	~~1~~	~~1~~ ②
P5	0	0	2	5	3	3	7	5	5	~~5~~	~~3~~	~~1~~ ③
							10, 5, 7					

(+P1, +P3, +P4, +P5)

① find P_i ⇒ Remain need <= Available ⇒
P2 ⇒ terminate P2 and update Available (Available + P_2 allocation)
⇒ (3 3 2) + (2 0 0) = 5 3 2

② P4 ⇒ terminate P4 and update available
(5 3 2) + (2 1 1) = 7 4 3

Safe sequence ⇒ $P_2 \rightarrow P_4 \rightarrow P_5 \rightarrow P_1 \rightarrow P_3$

Note: There may be various sequence, it depends on which process you terminate first.

★ Deadlock Detection

- Single Instanced
  - Detect cycle
- Multiple Instanced
  - Safety Algorithm

★ Recovery

- Resources
  - Preempt
  - Rollback
- Processes
  - Kill one
  - Kill All

UNIT: 2.1

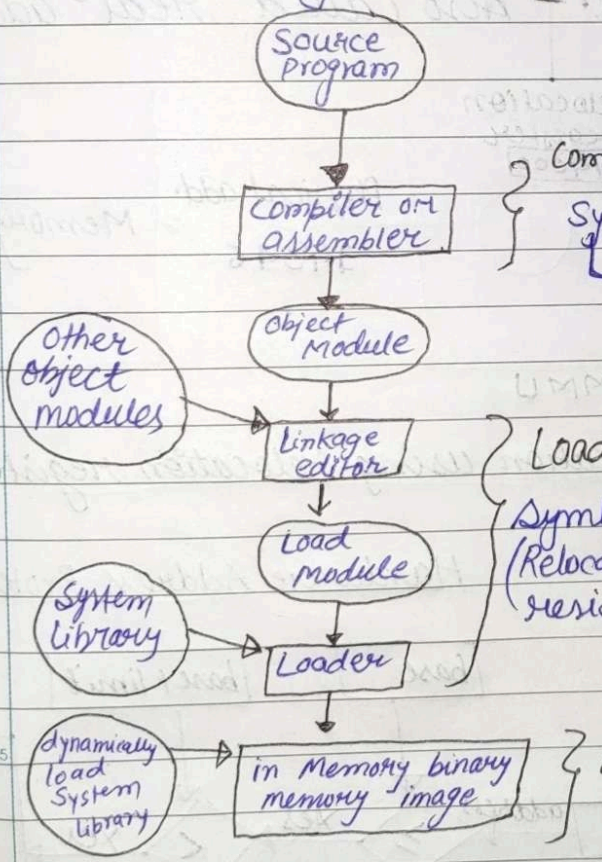
Dashrath Nandan

Date :

Memory Management

Memory management is the process of controlling and coordinating computer memory, assigning blocks to various running program to optimize system performance.

Address Binding is the process of mapping the program's logical or virtual addresses to corresponding physical or main memory addresses.



Multistep processing of a User Program.

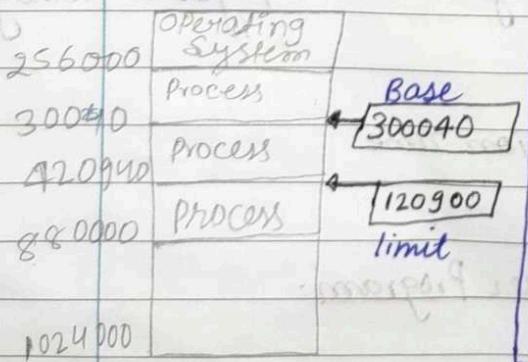
* Three types of addresses:
  i) Symbolic Addresses
  ii) Relative addresses (Relocatable)
  iii) Physical addresses

Date :

* Logical Address	Physical address Space
1) Generated by CPU.	– location in a memory unit.
2) Logically address space is set of all logical addresses generated by CPU.	– Physical address space is a set of all physical addresses mapped to the corresponding logical addresses.
3) User can view LA of a program.	– User can never view Physical add. of a program.
4) It can be change. Also called virtual add.	– It will not change. – Also called real address

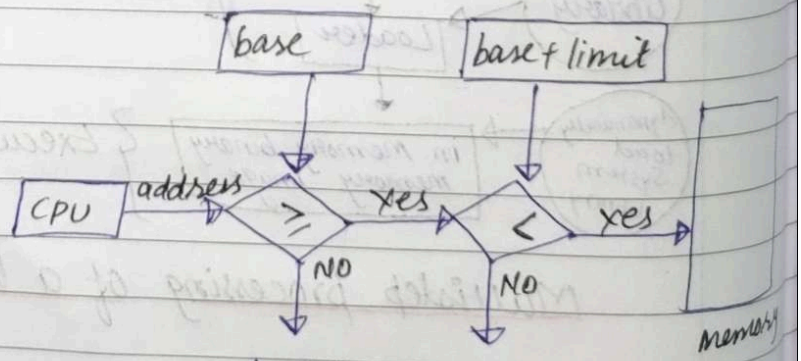
Dynamic relocation using relocation register.

* Base & limit Register



A pair of base & limit register define the logical add. space

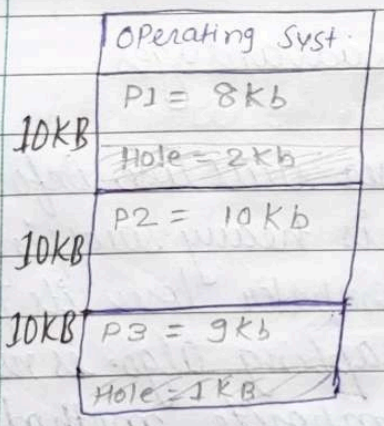
Hardware Address Protection



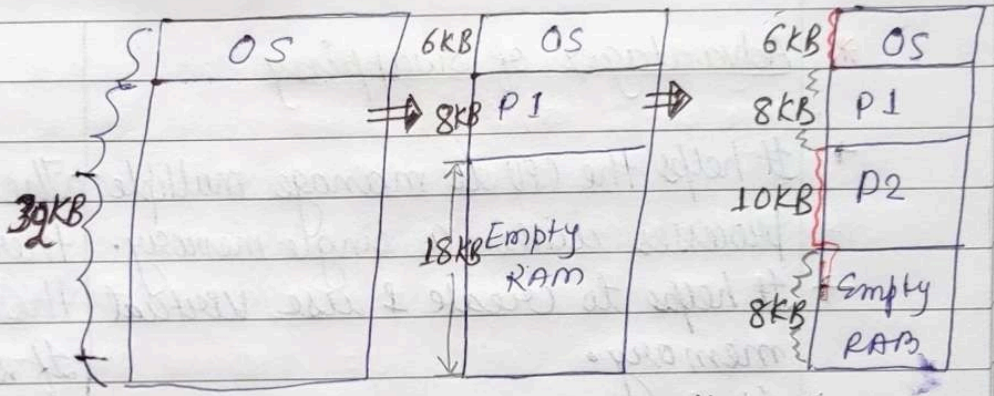
Memory Allocation

High memory (where User Processes are held) can be partitioned in following ways:

- Contiguous Allocation : a) Single/Fixed - Size partition
  b) Multiple/variable-Size partition
- Dynamic Storage Allocation : a) First Fit
  b) Best Fit
  c) Worst Fit



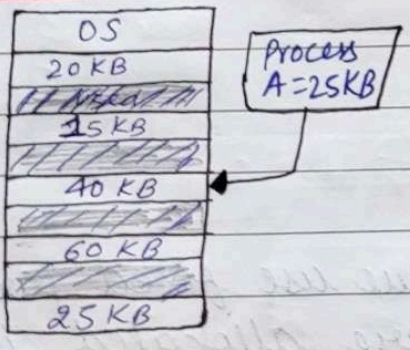
Single/Fixed partition



Variable-Size partition

→ Memory is partioned into similar or varible sizes. At runtime process request block, if memory is available, then process will allocated a block.

First Fit

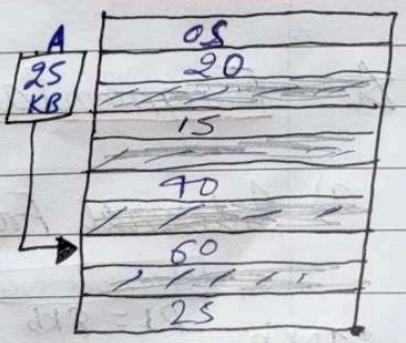


The first hole that is big enough is allocated to program.

Best Fit

The smallest hole that is big enough is allocated

Worst Fit

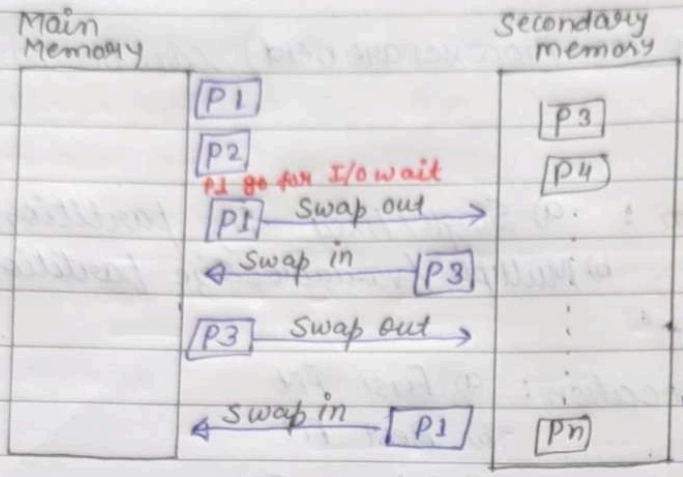


The largest hole that is big enough is allocated.

Date :

☆ Swapping:

Also known as a technique for Memory compaction.



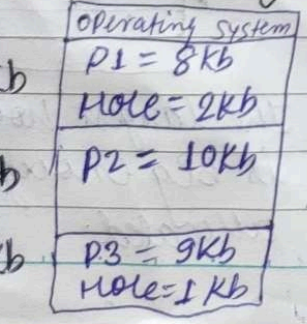
Swapping is a mechanism in which a process can be swapped out temporarily of main memory to secondary memory so that main m/m can be made available for other processes.

* Advantages of Swapping	Disadvantages
→ It helps the CPU to manage multiple processes within a single memory.	→ The user will lose info if there is heavy swapping and the computer loses its power.
→ It helps to create & use virtual memory.	→ If swapping algo. is not good, the composite method can increase the no. of page fault.
→ It improves main memory utilization	

* Swap Time : Questions / Numerical (PPt)

☆ Fragmentation:

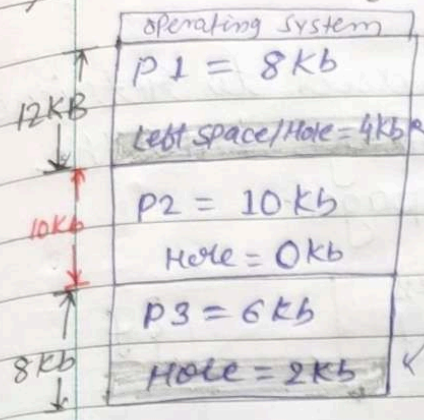
a) Internal Fragmentation:



→ It arises when we use fixed partitioning.
→ Here 2 processes are allocated space more than required and this unused space is so small to store a new process and wasted. This is called Internal fragmentation.

Date :

b) External Fragmentation :

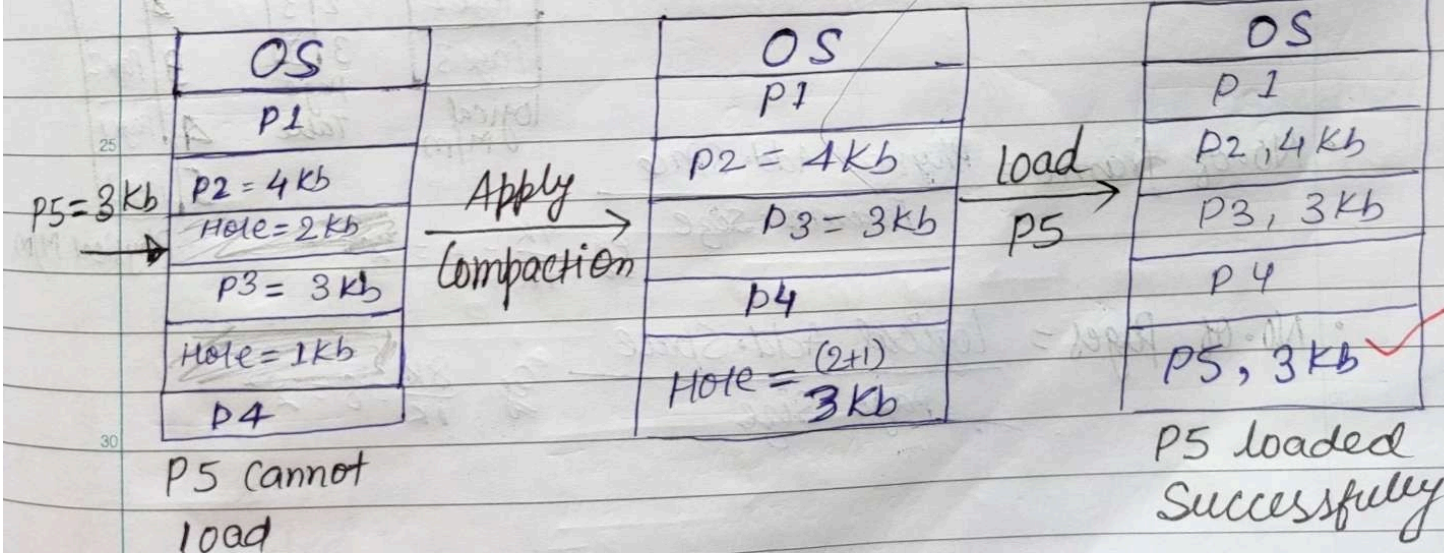


⇒ It arises when dynamic partitioning technique is used.

⇒ Now, if a new process (P4 = 6Kb) wants to be swapped. Though we have total 6Kb space but we cannot service this request as these blocks are not Contiguous (Adjacent). This is called External fragmentation.

☆ Compaction

- Compaction is a process in which the free space is collected in a large memory chunk to make some available for processes.
- It helps to solve the problem of fragmentation, but it requires too much of CPU time.



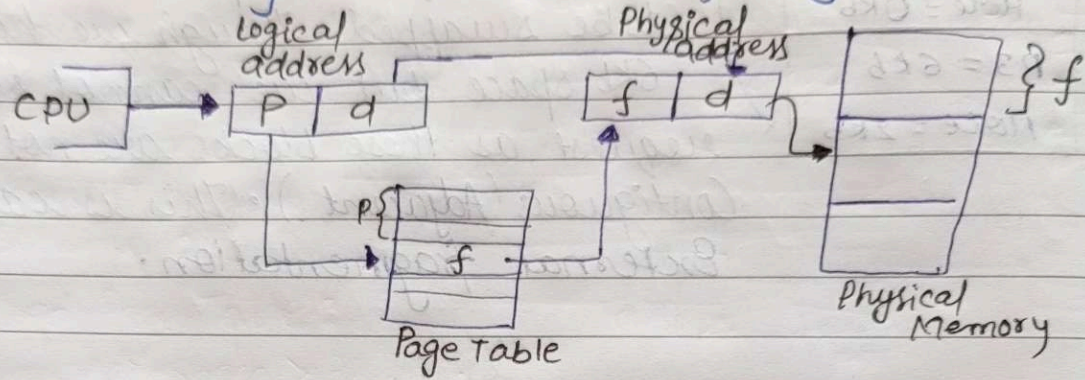
P5 cannot load

P5 loaded Successfully

Date :

☆ Paging

Paging is a storage mechanism used to retrieve processes from the secondary storage into the main memory in the form of pages.

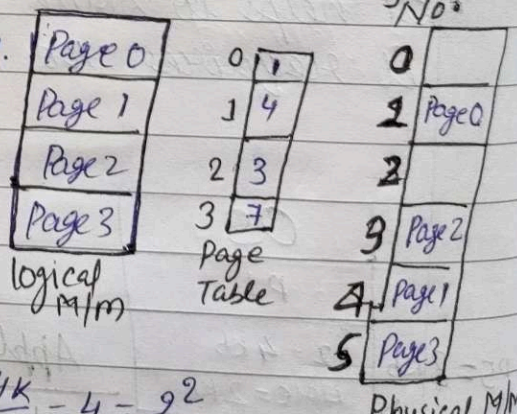


Page Number (p): Used as an index into page table.

Page offset (d): Combined with base add. to define the physical add. that is sent to the memory unit.

* Address Translation Scheme:
  - Logical address = Page no. + page offset (d)
  - Physical address = Frame no. + Page offset (d)

→ Physical M/M generates Frames.
→ Logical M/M generates Pages.



- No. of frame = $\frac{\text{Physical Add. Space}}{\text{Frame Size}}$ Eg. $\frac{4K}{1K} = 4 = 2^2$
- No. of Pages = $\frac{\text{Logical Add. Space}}{\text{page Size}}$ Eg $\frac{8K}{1K} = 8 = 2^3$

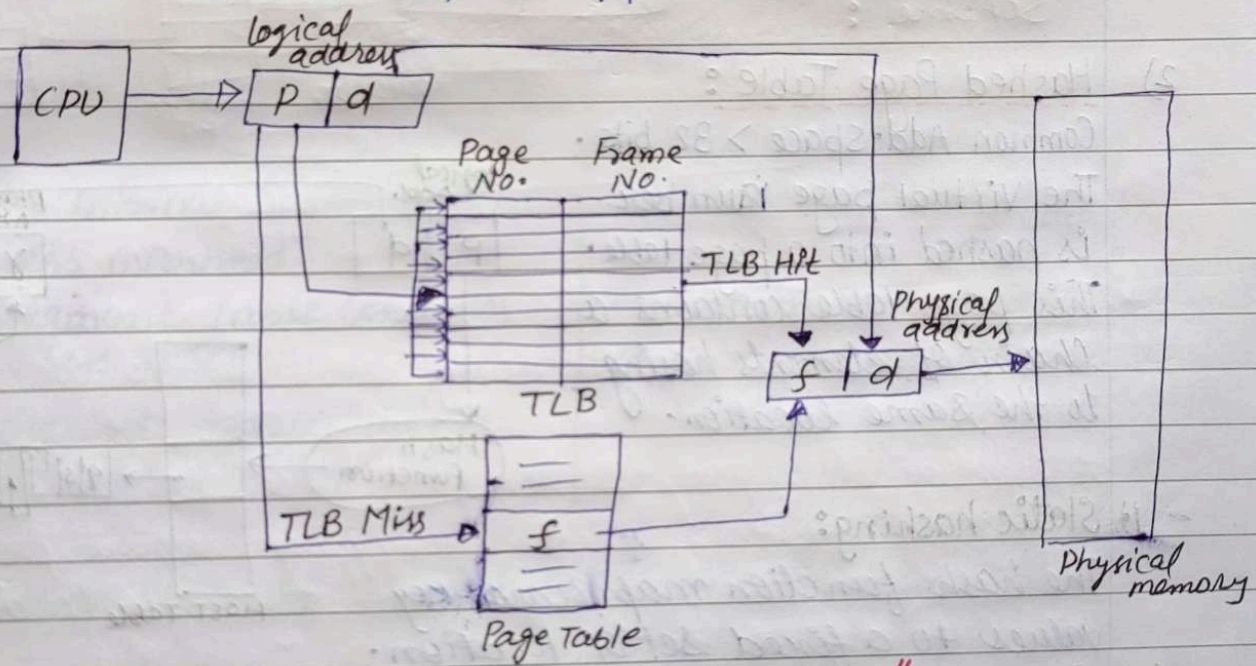
Effective Access Time = (Hit ratio of TLB) * (Access Time of TLB + A.T. of main mem)
+
(Miss ratio of TLB) x (AT of TLB + 2 x AT of Main mem)

Date : _____

☆ Paging with TLB

Translation Look-Aside Buffer, a table in the processors memory that contains information about the pages in the memory processor has accessed recently.

- The TLB enables faster computing because it allows the address processing to take place independent of the normal address-translation pipeline.



"Paging Hardware with TLB."

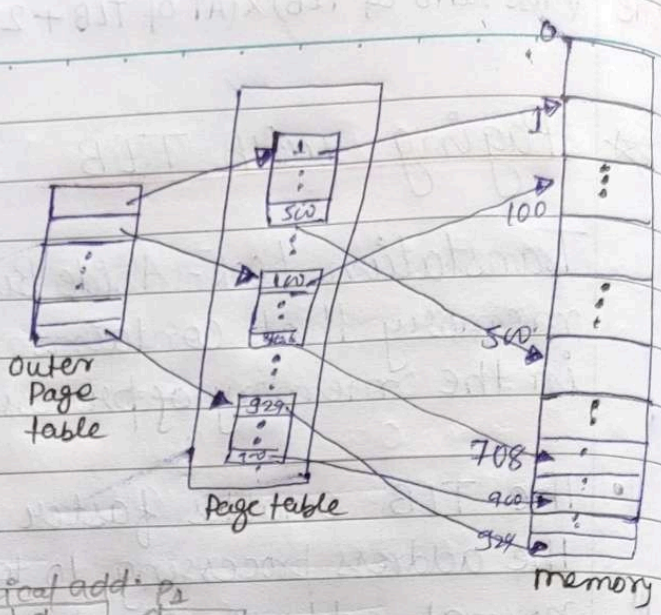
↳ Page-table base register (PTBR) points to the page table.
↳ Page-table length register (PTLR) Indicates size of page table.

* Page Table structure

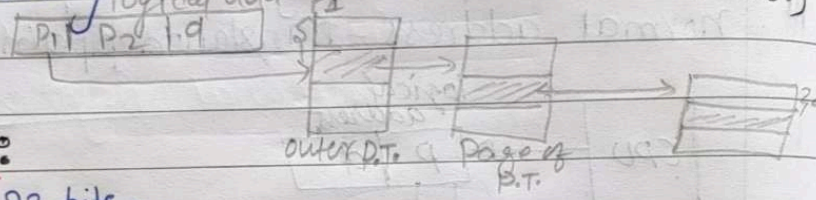
1) Hierarchical Page Table.
2) Hashed page table.
3) Inverted page table.

Date :

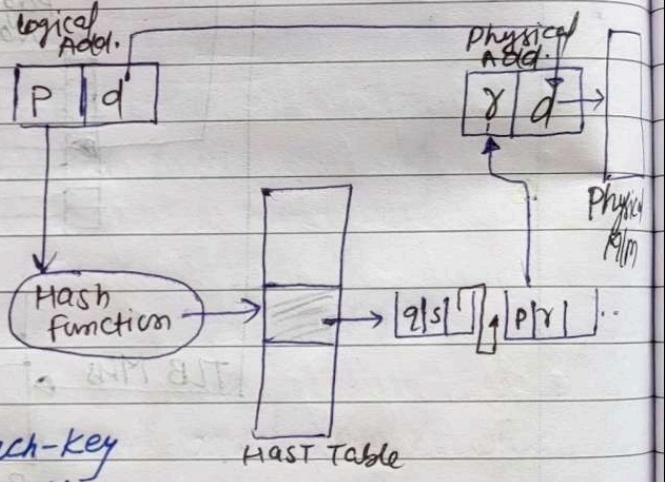
(1) Hierarchical page Table:
Multilevel paging is a type of paging where logical address space is broken up into multiple page table when the page table size cannot fit into the memory.



Address-Translation Scheme :



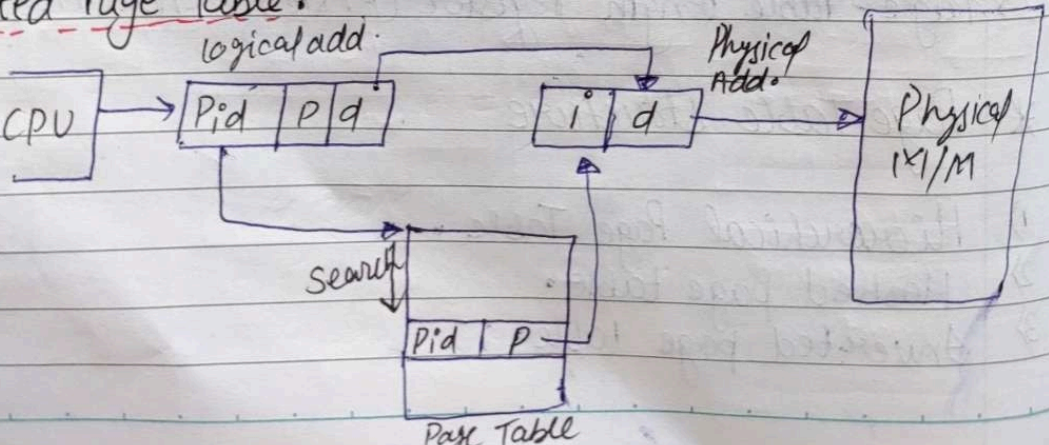
2) Hashed Page Table:
Common Add. Space > 32 bits.
The virtual page number is hashed into a page table. This page table contains a chain of elements hasing to the same location.



1) Static hashing:
The hash function map search-key values to a fixed set of location.

2) Dynamic hashing: A hash table can grow to handle more items.

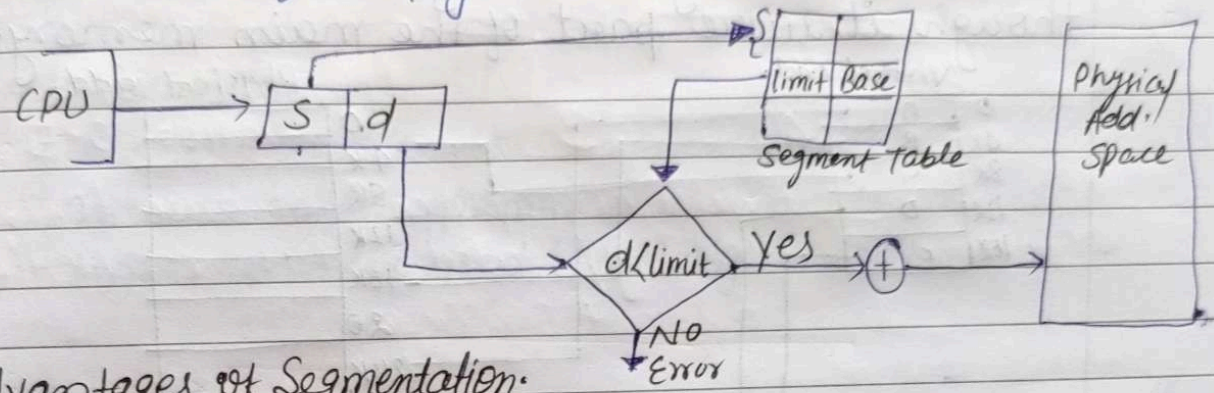
3) Inverted Page Table:



Date :

☆ Segmentation

In Segmented Paging, the main memory is divided into variable size segments which are further divided into fixed pages.



* Advantages of Segmentation.

i) No internal fragmentation.
ii) Less overhead.
iii) Segment Table Consumes less Space.
iv) Flexible Segment Size.

Disadvantage

→ It can have external fragmentation.
→ Costly m/m management algo.

* Advantages of Paging

i) Efficient use of memory.
ii) Protection and Security.
iii) Reduce fragmentation.
iv) Virtual memory.

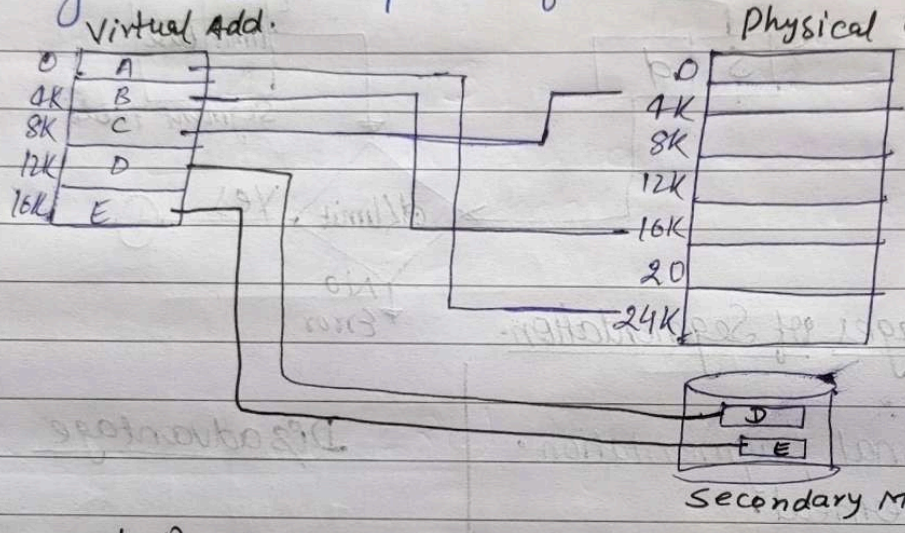
Disadvantages

→ Overhead and latency.
→ Increased Complexity.
→ Memory thrashing.

Date :

☆ Virtual Memory

Virtual Memory is a storage allocation scheme in which secondary memory can be addressed as though it were part of the main memory.



* Demand Paging:

Virtual memory is commonly implement by demand paging.

A demand paging system is quite similar to a paging system with swapping where process resides in secondary m/m and pages are loaded only on demand.

- Page Fault: A page fault occurs when an invalid page is addressed.

- Effective memory access time = $(p * s) + (1-p) * m$

  p: page fault rate
  s: page fault service time.
  m: main m/m access time

Date :

☆ Page Replacement

When the main memory fills up, a page must be swapped out to make room for any page to be swap in.

Page replacement algorithm –

i) First In First Out (FIFO) algorithm.
ii) Optimal Page algorithm
iii) Least Recently Used (LRU) algorithm.

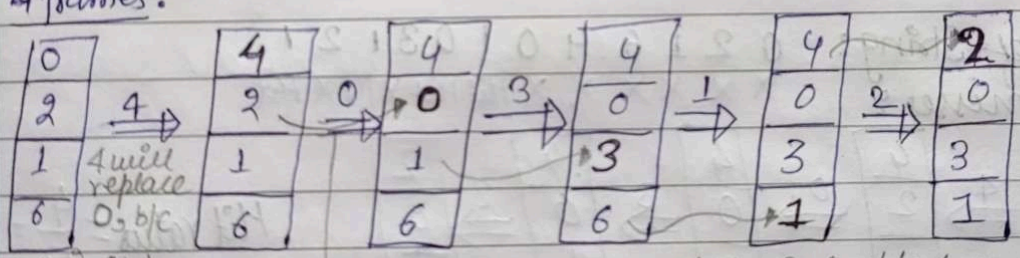
1) First In First Out :

Oldest page in the memory is the one will be replaced.
(जो पहले आयेगा वो replace होगा).

Eg: Ref. string: 0 2 1 6 4 0 1 0 3 1 2 1

Misses: X X X X X X Hit Hit X X X Hit

4 frames:



2 will be replaced b/c 2 is oldest among (4,2,1,6)

$$\text{Fault Rate} = \frac{9(\text{miss})}{12} = 0.75$$

$$\text{Hit Ratio} = 3/12 = 0.25$$

☆ Optimal Page Replacement (Page demand after long time).

It replaces the page that will not be used for the longest period of time.

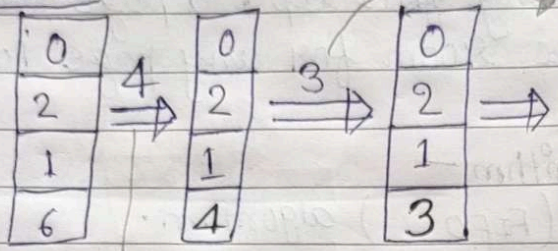
Date :

Eg

Ref. String: 0 2 1 6 4 0 1 0 3 1 2 1

misses : × × × × × Hit Hit Hit × Hit Hit Hit

4 frames:



we can replace either 0/4 because both are not needed in future.

Hit ratio = 6/12 = 0.5

Miss ratio = 0.5

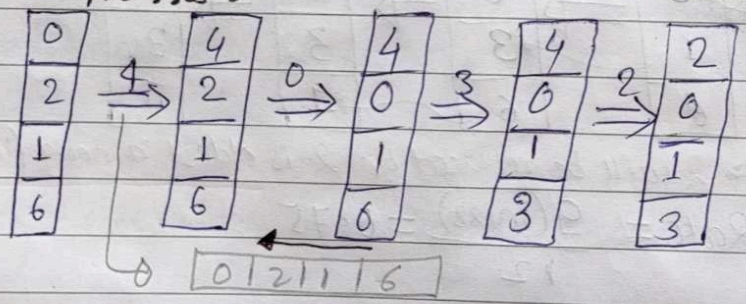
6 will be replaced because (0,2,1) are coming/needed in future after '4' but 6 is not coming.

(3) Least Recent Use (LRU) Page Replacement

LRU choose the page that has not been used for the longest period of time in the past.

Eg Ref. String: 0 2 1 6 4 0 1 0 3 1 2 1

misses: × × × × × × Hit Hit × Hit × Hit



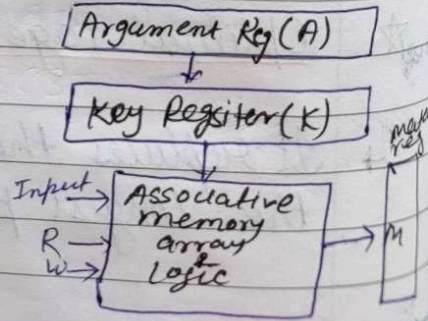
Hit ratio = $\frac{4}{12} = \frac{1}{3}$

Miss ratio = $\frac{8}{12} = \frac{2}{3}$

4 will replace '0' because it used a long time in the past.

☆ Associative Memory:

It is often reffered as Content Addressable Memory (CAM).



Date :

Memory is accessed simultaneously and in parallel on basis of data rather than address or location.

☆ Thrashing: It is a phenomenon that occurs in computer system when the system spends an excessive amount of time on page swapping rather than executing useful work.

Techniques to handle Thrashing.

i) Working set model

ii) Page fault frequency

UNIT: 2.2 Dashrath Nandan Date :

☆ Device Management

Device management in OS manges all the I/O devices such as keyboard, disk, printer, etc.

* The I/O devices can diuided into 3 categories:

i) Block device: It stores info in fixed-size block, eg Disk.
ii) Character device: delivers or accept a stream of characters, eg-Printer, Keyboard
iii) Network device: For transmitting data packets.

* The OS peripheral devices can be categorized into :-

i) Dedicated device: Dedicated to only one job at a time Printer
ii) Shared devices: These can be allocated to several processes. eg. Harddisk.
iii) Virtual devices: Combination of dedicated & Shared devices

* Magnetic Disk: Magnetic disk provide the bulk of storage for modern computer System.

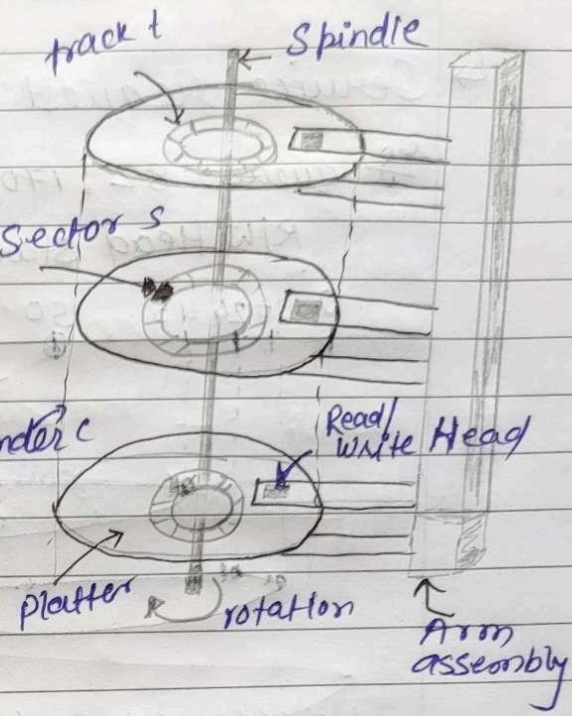
- Different types of Disk:

i) Advanced Technology Attachment (ATA) eg: Hard drive, CD ROM
ii) Small Computer System Interface (SCSI) → allow recording
iii) SATA II (Serial ATA)

Date : _____

☆ Physical Disk Structure:

A read-write head "flies" just above each surface of platter. The head are attached to a disk arm that move all the head as a unit. Platter is divided into circular tracks which are subdivided into sectors.



* Disk Management:

Things to do in Disk management –

⇒ Partition a Drive ; Formate a Drive; Shrink a Drive; Delete a partition; Change a Driver's File System.

* Disk Access Time: Two major Component

↳ Seek time: It is the time for disk to move the heads to desired sector.

↳ Rotational latency: Time to rotate disk to desired sector.

Imp

☆ Disk Scheduling:

Disk bandwidth is the total no. of bytes transferred, divided by the total time between the first request for service and completion of last transfer.

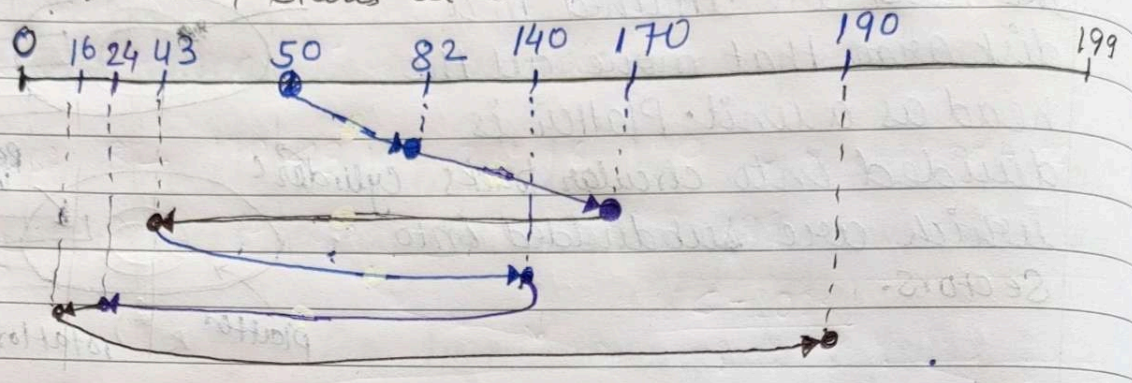
Date :

1) FCFS (First Come First Serve)

↳ Service request in the order they come.

Eg: Queue: 82, 170, 43, 140, 24, 16, 190

R/W Head starts at 50.



No. of movements = (82-50) + (140-82) + (170-43) + (140-43) + (140-24) + (24-16) + (190-16) = 642 Ans

2) Shortest Seek Time first (SSTF):

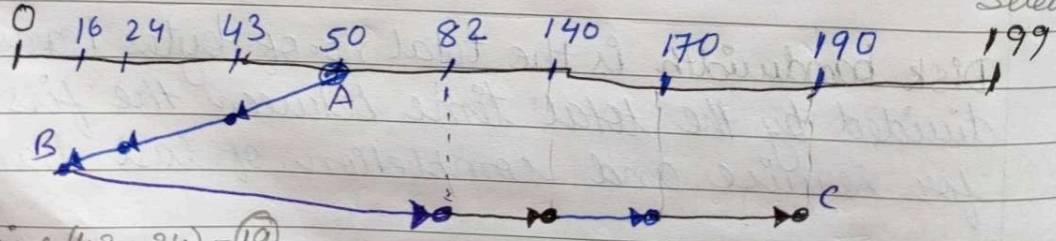
↳ Select the request with minimum seek time from the current head.

Eg: Queue: 82, 170, 43, 140, 24, 16, 190

R/W Head: 50

Seek time (50-82) = 32

Seek time (50-43) = (7) Select



Seek time (43-24) = (19) → Select as next request as its seek time is less.

" " (43-82) = 39

No. of movements = dist(AB) + (BC)

= (50-16) + (190-16) = 208

Date :

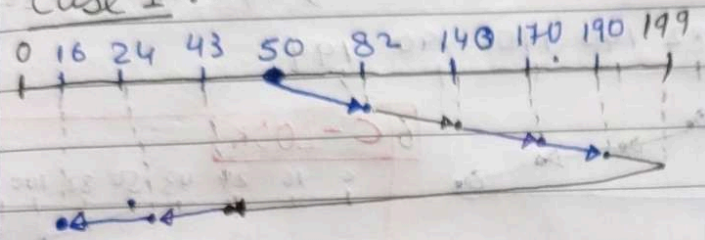
3.) SCAN

The disk arm starts towards one end [either lower or higher] servicing request until it get reached the end, where the head movement is reversed and servicing continues.

Eg Queue : 82, 170, 43, 140, 24, 16, 190

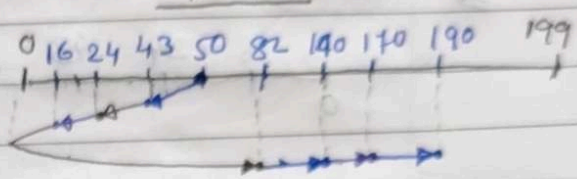
R/W Head : 50

Case I :



No. of Movements = (199-50) + (199-16)

Case II :



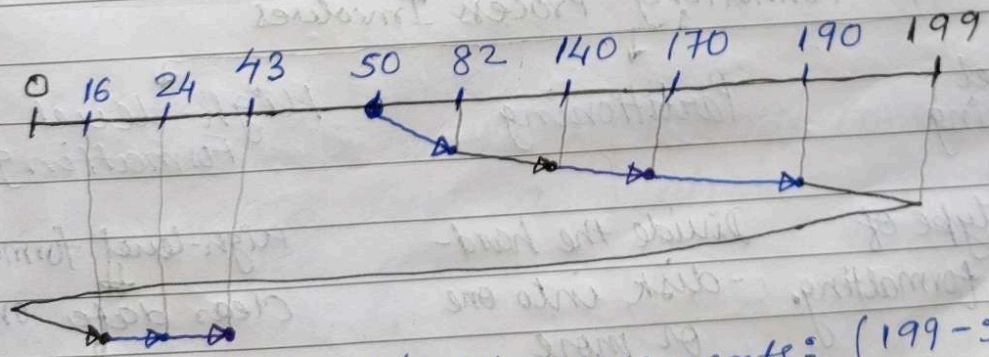
No. of movements = (50-0) + (190-0) = 240

4.) C-Scan [Circular Scan]

The head moves from one end to other servicing all request. When it reaches the other end, it immediately returns to the beginning to other end without serving request, then starts again from beginning of disk.

Eg: Queue : 82, 170, 43, 140, 24, 16, 190

R/W - Head : 50



No. of movements : (199-50) + (199-0) + (43-0)

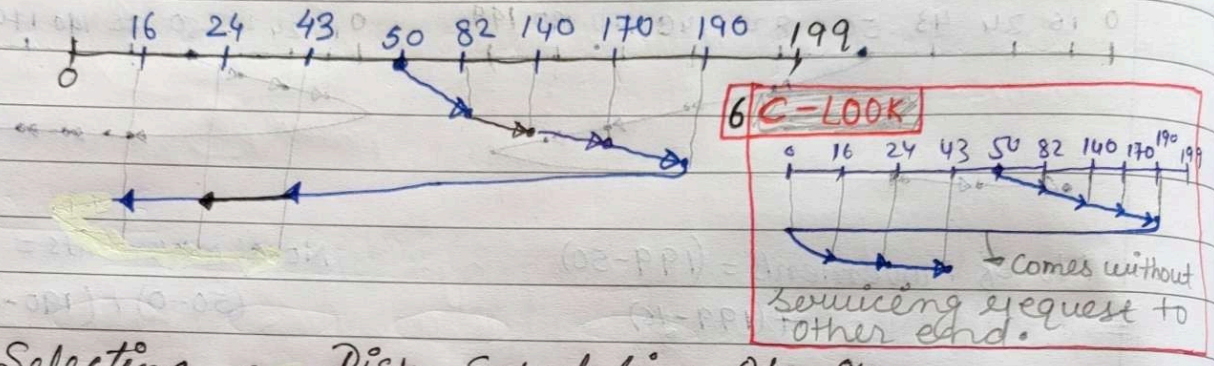
Date :

5.) LOOK SCAN:

Version of C-SCAN.

Arm only goes as far as the last request in each direction, then reverse direction immediately, servicing request to other end till the last service is requested.

Eg Queue: 82 170 43 140 24 16 190 , R/w-Head = 50



* Selecting a Disk-Scheduling Algorithm

i) SSTF is common and has natural appeal.
ii) SCAN & C-SCAN perform better for system in heavy load.
iii) Either SSTF or C-LOOK is reasonable choice for default algo.

☆ Disk Formatting

Disk Formatting is a process to configure the data-storage devices such as hard-drive, etc. when we are going to use them for the very first time or inital usage.

Disk-Formatting Process Involves

Low level Formatting	Partitioning	High-level Formatting
→ It is a type of Physical formatting.	Divide the hard-disk into one or more region	High-level formatting ↓ Clear data on hard disk ↓ Generate boot info ↓ Initialize FAT ↓ Label logical bad Sectors

Date :

☆ RAID (Redundant Array of Independent Disk)

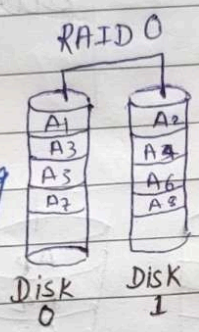
The basic idea of RAID is to store the same data in different places (thus, redundantly) on multiple hard-disk.

↳ Increase data reliability & capacity
↳ Increase I/O performance.

The Different RAID levels

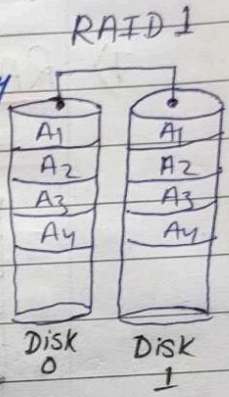
1) RAID 0

The data is split across drivers, resulting in higher data throughput.

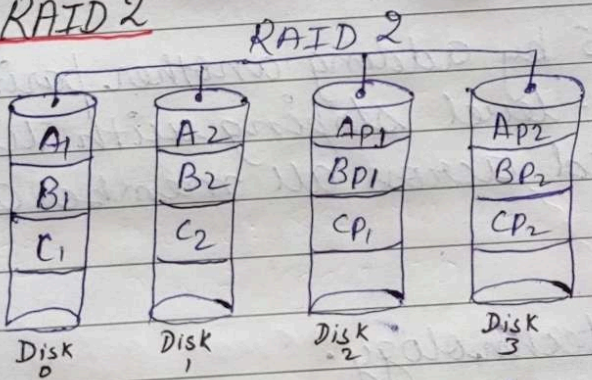


2) RAID 1

It provides redundancy by writing all data to two or more drivers. This level reffered as disk mirroring.

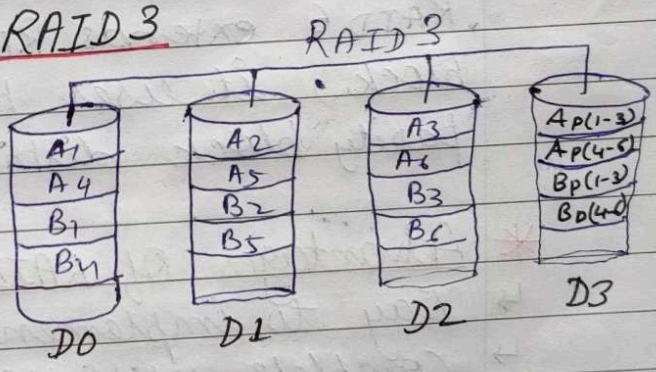


3) RAID 2



RAID 2, which is rarely used, strips data at the bit level, and uses a Hamming code for error corrections.

4) RAID 3

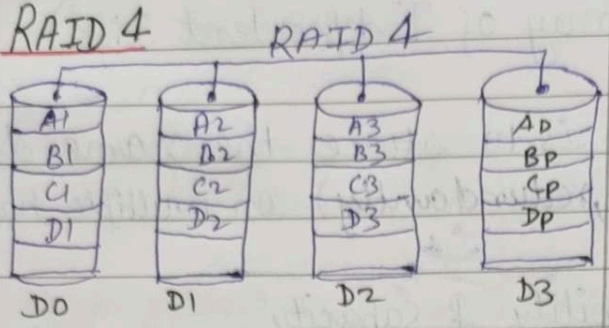


RAID 3, consist of byte level striping with a dedicated parity disk.

- Striping is the process of dividing a body of data into blocks and spreading data blocks to multiple storage.

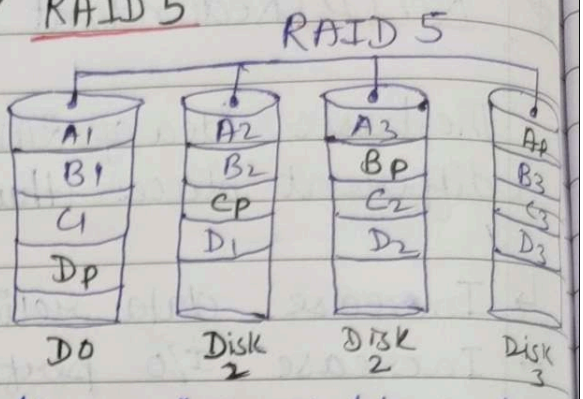
Date :

5) RAID 4



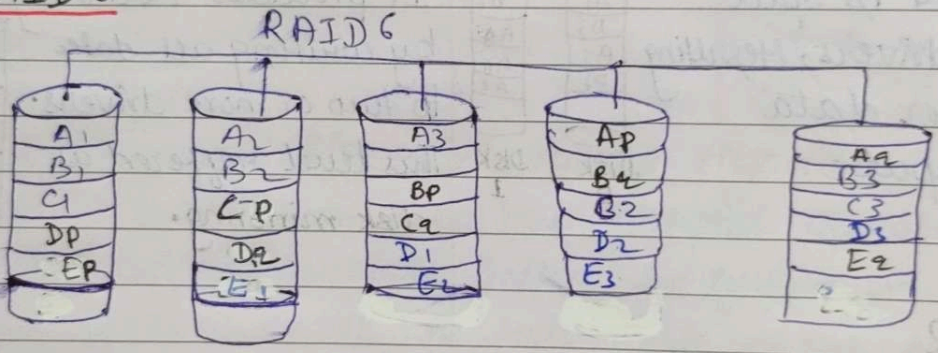
RAID 4 consist of block-level striping with a dedicated parity disk.

6) RAID 5



It consist of block-level striping with distributed parity.

7) RAID 6



RAID 6 extends RAID 5 by adding another parity block, it uses block-level striping with two parity block distributed across all member disk.

Advantages of RAID:

i) Increased data reliability.
ii) Improved performance.
iii) Scalability
iv) Cost-effective.

Disadvantages of RAID:

i) Complexity.
ii) Increased Risk of data loss.
iii) Some RAID configurations are costly.

UNIT: 2.3 Dashrath Nandan Date :

☆ File Management :

The process and act of creating an organized structure in which you store info. for easy retrival.

File: A file is a logical unit of information created by processes and managed by OS.

* File Attributes

i) Name ii) Identifier (unique tag) iii) Type iv) Location
v) Size vi) Protection (Access control) vii) Time, date

* File Operations

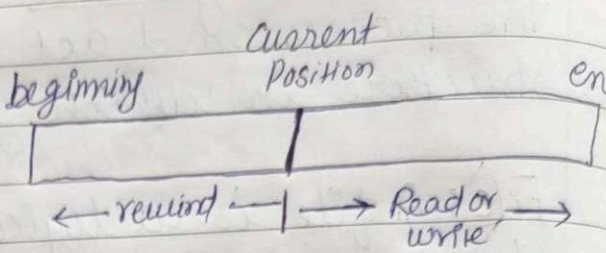
i) Writing a file ii) Reading a file iii) Deleting a file
iv) Repositioning within a file — directory is searched for the appropriate entry, & current-file-position pointer is repositioned to a given value. This file operation is known as File Seek.
v) Truncating a file :- Erase the content but kept its attributes.

* File Type	Usual extensions	Function
• executable	exe, com, bin or none	Ready-to-run machine language program.
• object	obj, o	Compiled, machine language, not linked
• text	txt, doc	textual data, document
• Source code	c, cc, java, pas	Source code in various languages.

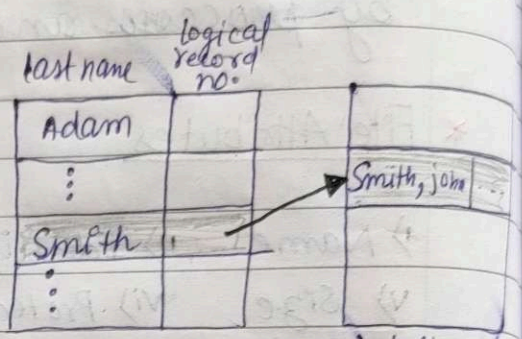
Date :

★ File Access Method : → Sequential Access
→ Direct Access

i) Sequential Access :
Information in file is processed in order, one record after the other.



ii) Direct Access :
It relies on addressing techniques that enable the operating system (OS) to identify the data's location without having to search for the data



Example of Index & Relative files.

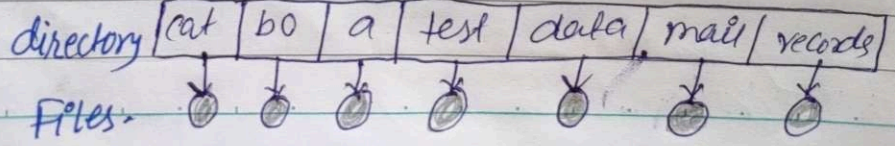
iii) ISAM (Indexed Sequential Access Method)
Modification of Direct access. Combination of both the sequential and direct access.

★ Directory
A directory is a container that is used to contain folders and files

* Operation : i) Search for a file ii) Create a file
iii) delete a file iv) list a directory v) Rename a file

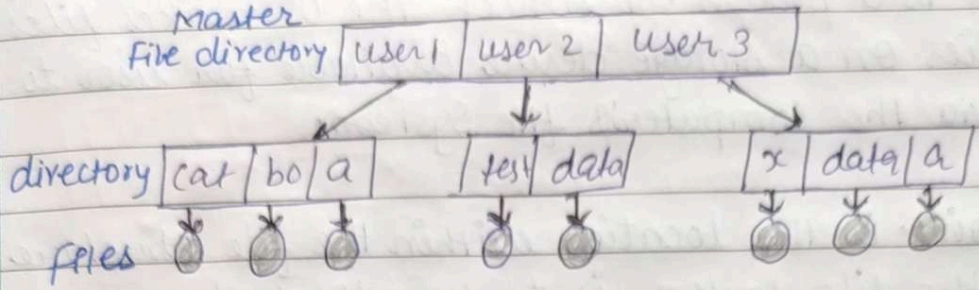
* Structure of Directories :

1) Single-level Directory : A single directory for all user.



Date :

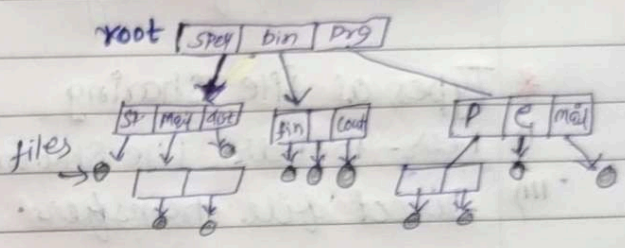
2) Two-level Directory : Seperate directory for each user.



3) Tree-Structured Directories:

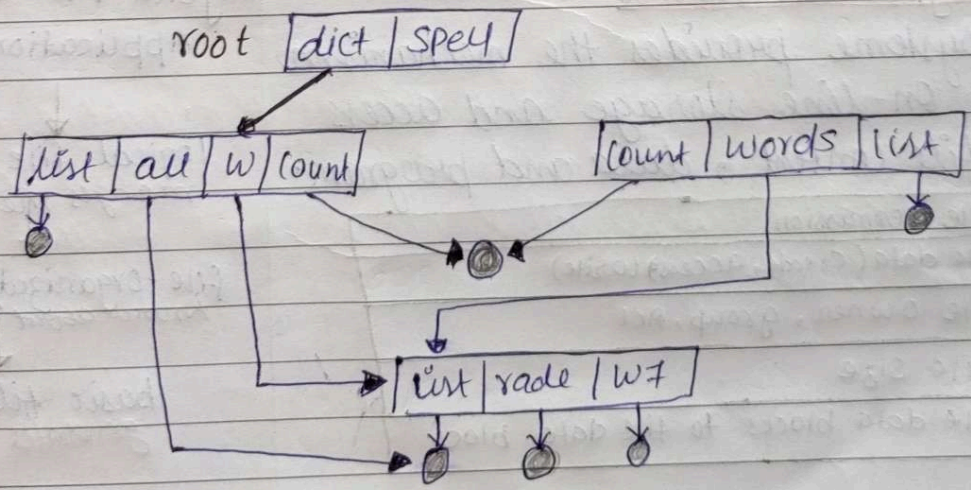
↳ Efficient Searching

↳ Grouping Capability



- absolute Path name begins at the root, follows a path down to specified file.
- A Relative path name defines a path from the current directory.

4) Acyclic-Graph Directories:
Have shared subdirectories and files.



Date :

☆ File Sharing :

Mounting is a process by which the OS makes files and directories on a storage device available for user to access via the computer's file system.

Mount point is the location within the file structure where the file system is to be attached.

* Types of file sharing
i) Peer-to-Peer file sharing ii) Cloud based file sharing
iii) Direct file transfer.

☆ Protection

Protection mechanism provides controlled access by limiting the type of file access that can be made.

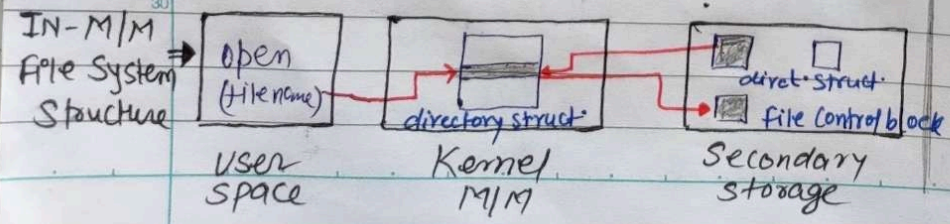
Type of Access : i) Read ii) Write iii) Execute iv) Append
v) Delete vi) List

☆ File System Structure

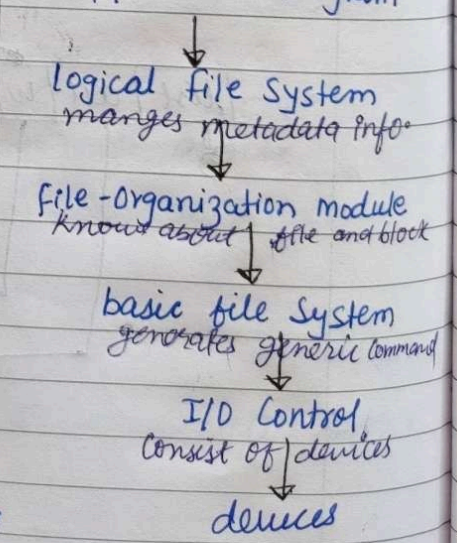
File systems provides the mechanism for on-line storage and access to file content, data and program.

File Control Block

File permission
file data (create, access, write)
file owner, group, ACL
file size
file data blocks to file data block



Layered File System



Date :

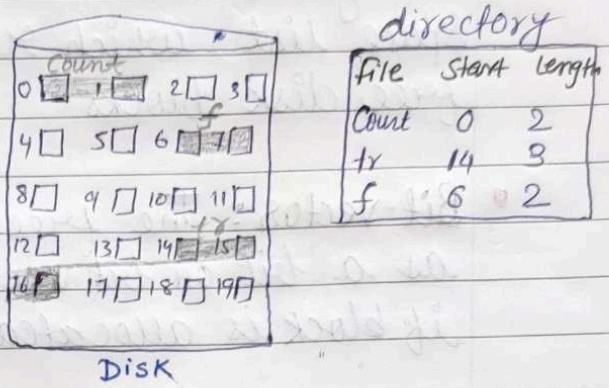
☆ Allocation Methods

An allocation method refers to how disk blocks are allocated for files.

1) Contiguous Allocation:

Contiguous allocation requires that each file occupy a set of contiguous blocks on the disk.

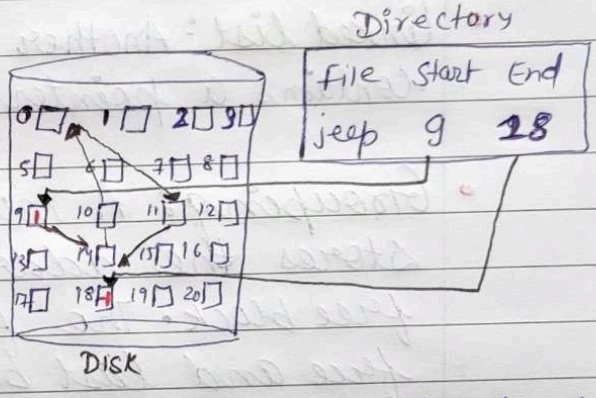
↳ One difficulty is finding space for a new file.

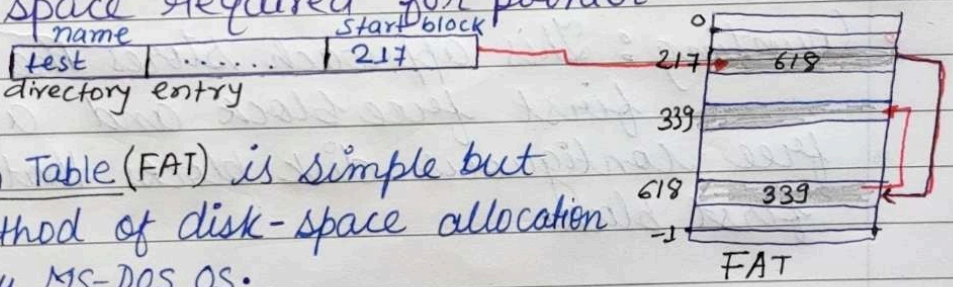


2) Linked Allocation:

It solves all problems of contiguous allocation. With linked allocation, each file is a linked list of the disk block. The disk may be scattered anywhere on disk.

↳ It is inefficiento for direct access, and another disadvantage is the space required for pointer.

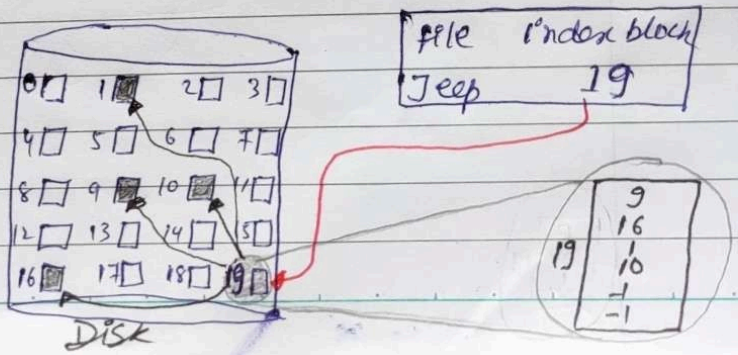




File Allocation Table (FAT) is simple but efficient method of disk-space allocation was used by MS-DOS OS.

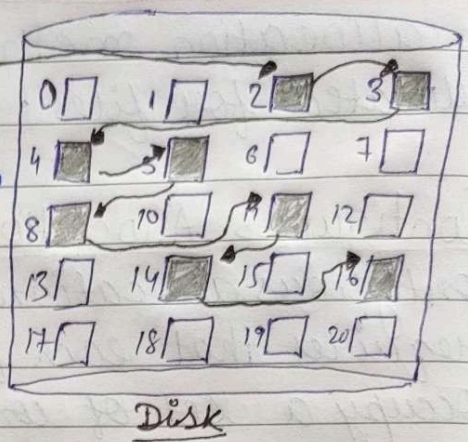
3) Indexed Allocation:

Each file has its own index block, which is an array of disk-block addresses.



Date :

☆ Free-Space Management



To keep track of free disk space, the System maintains a free-space list, which records all free disk blocks.

- **Bit-vector**: The free-space list is implemented as a bit or bit vector. If block is free, bit is 1. if block is allocated, bit is 0.

- **Linked list**: Another approach, where first block contains a pointer to the next free disk block.

- **Grouping**: A modification of free-list approach, stores the addresses of n-free blocks in first free block. The first $n-1$ of these blocks are actually free and last block contain the addresses of another n free blocks.

- **Counting**: This approach stores the address of the first free block and a number n of free contiguous disk blocks that follow the first block.

Date :

Numericals [Word file, PPT or YT]

1) Problems on memory partitions.

2) Problems on Paging, Paging with TLB.

3) Problems on Segmentations

4) Problems on Page replacement algos.
   - Hit ratio
   - miss ratio.

5) DISK Scheduling.

UNIT: 3

Dashrath Nandan

Date :

☆ Protection refers to a mechanism which control the access of programs, processes or user to the resources defined by a computer System.

Need of Protection:

i) To prevent the access of unauthorized user.
ii) To ensure that each active programs or processes in the system uses resources only as stated policy.
iii) To improve reliability by detecting latent errors.

Role of Protection:

The role of protection is to provide a mechanism that implements policies which define the uses of resources in computer system.

Guiding Principle -

i) Programs, users and System should be given just enough Privileges to perform their task.
ii) Limits damage if entity has a bug.
iii) Must consider "Grain" aspects -
- Rough-grained privilege management easier, simple.
- Fine-grained privilege management more complex, more overhead but more protective.

* Domain Structure:

- Access-right = <object-name, right set>; where right-set is a subset of all valid operations than can performed on the object.
- Domain = Set of access-rights

D1
<O3, {Read, Write}>
<O1, {R, W}>
<O2, execute>

Date :

★ Access Matrix

Access Matrix is a security model of protection state in computer. It is used to define the rights of each process executing in the domain with respect to each object.

- Row represent domains.
- Column represent objects.
- Access(i,j) is the set of operation that a process executing in Domain$_i$ can invoke on object$_j$.

Obj / Domain	F_1	F_2	F_3	Printer
D_1	R		R	
D_2	R, W			Print
D_3		R/W	execute	

Implementation of Access Matrix

1. Global table:
   - Store ordered triples <domain, obj, right-set> in table.
   - But table could be large → won't fit.
   - Difficult to group objects.

2. Access lists for objects:
   - Each column implemented as an access list for one object.
   - Each column = Access control list, who perform what operation.
   - Each Row = Capability list (like a key).

3. Capability lists for domains:
   Instead of object-based, list is domain based. A subject is allowed to access any object for which it holds capabilities.

Capability Format

Object Descriptor	Rights of the Subject

Capability list for domain is list of objects together with operations allows on them

Date :

④ Lock & Key:

The lock and key method is an hybrid of the access control or list and capabilities method.

↳ Each object has list of unique bit pattern, called locks.
↳ Each domain has list of unique bit pattern, called Keys.
Process in a domain can only access object if domain has key that matches one of the lock.

★ Security

Security violations (or misuse) of the system can be categorized as malicious or accidenta.

Types of Security violations :

i) Breach of Confidentiality : Unauthorized reading of data.
ii) Breach of Integrity : Unauthorized modification of data.
iii) Breach of availability : Unauthorized destruction of data.
iv) Theft of Service : Unauthorized use of resources.
v) Denial of Service (DOS) : Prevention of legitimate use :

Security Violation Methods:

i) Masquerading (breach authentication) - Pretending to be an authorized user to escalate privileges.
ii) Replay attack - As is or with message modification.
iii) Man-in-middle attack - Intruder sits in data flow.
iv) Session hijacking - Intercept an already-established session to bypass authentication.

Date :

Security Measure Levels:

Security must occur at four levels to be effective:

i) Physical : Data Centres, Servers, Connected terminals.
ii) Human : Avoid Social engineering, phising
iii) Operating System : Protection mechanisms, debugging.
iv) Network : Intercepted communication, DOS.

★ Threats : Threat is potential Security violation

* Program Threats:

1. Trojan Horse : A code segment that misuses its environment is called a Trojan horse. It exploits mechanism for allowing program written by user to be executed by other users.

2. Trap Door : Specific user identifier or password that circumvents normal security procedures.

3. Logic Bomb : Program that initiates a security incident under certain circumstances.

4. Stack and Buffer Overflow : Exploits a bug in program.
   - ↳ Failure to check bound on inputs, arguments.
   - ↳ Unauthorized user or privilege escalation.

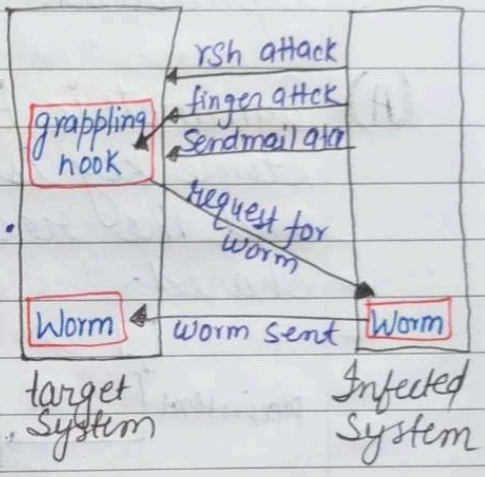
5. Viruses : A virus is a fragment of code embedded in a legitimate program. Viruses are self-replicating and are designed to "infect" other program.
   - File : A standard file viruses infect a system by appending itself to a file.

Date :

- Boot : A boot virus infects the boot sector of the system, it watches for other bootable media and infect them.
- Macro : Macro viruses are written in high-level language, are triggered when a program capable of executing the macro is run.
- Source code : A source code virus looks for source code and modifies it to include the virus & spread it.
  ↳ Polymorphic virus, Encrypted virus, Armored viruses, etc.

* System and Network Threats :

1. Worms : A worm is a process that uses the spawn machanism to duplicate itself. Grappling hook program uploaded main worm program. Hooked system then uploaded main program code, tried to attack connected systems.

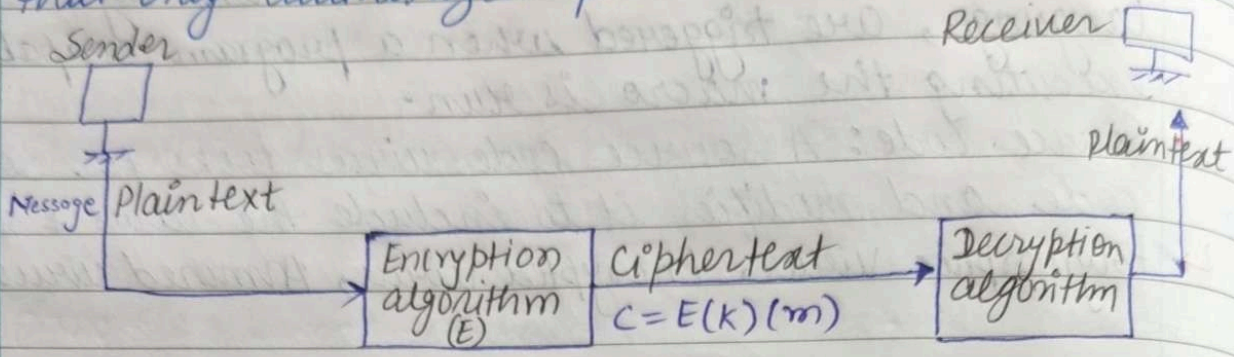


2. Port Scanning : Port scanning is not an attack but rather a means for a cracker to detect a system's vulnerability to attack. It is automated involving a tool that attempts to create a TCP/IP connection to a specific port or a range of port.

3. Denial of Services (DOS) : Overload the targeted computer preventing it from doing any useful work. It is impossible to prevent denial-of-services attacks. Distributed denial-of-services (DDOS) comes from multiple sites at once.

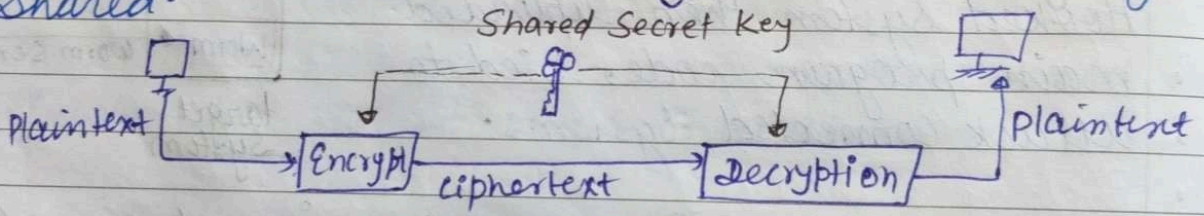
Date :

★ **Cryptography**: Cryptography is a technique of securing information through use of code so that only authorized person can understand it.

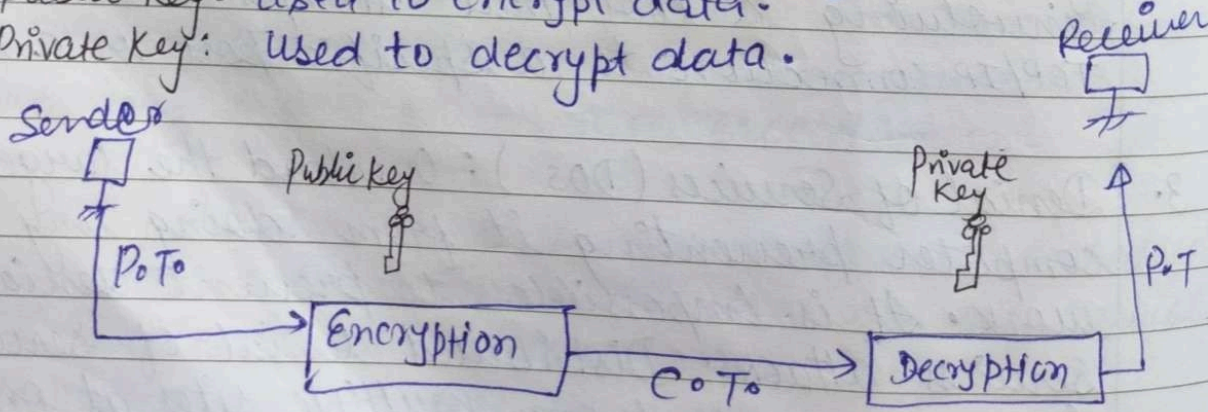


Encryption algorithm consists of –
↳ Set K of keys. ↳ Set M of messages ↳ Set C of ciphertexts.

(A) **Symmetric Encryption**: In Symmetric encryption same key is used by the sender (for Encryption) and the receiver (for decryption). The key is shared.



(B) **Asymmetric Encryption**: Public key encryption based on each user having two keys –
i) public key: Used to encrypt data.
ii) Private key: Used to decrypt data.



Date :

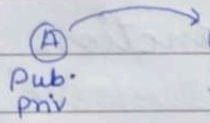
Assymmetric encryption algorithm is RSA, which is based on two large prime numbers, P and Q (on the order of 512 bit each), and their product N.

* RSA Algorithm :

1. Choose two different large random prime no; P and Q.
2. Calculate $n = p * q$
3. Calculate $\phi(n) = (p-1) * (q-1)$
4. Choose 'e' such that $1 < e < \phi(n)$,
   e is co-prime to $\phi(n)$, $gcd(e, \phi(n)) = 1$. $1 < e < 192$
5. Calculate d, such that $de \bmod \phi(n) = 1$.
6. public key 'e' private key 'd'.

Ex: A: P = 13, Q = 17. If the public key of A is 35. The private key?
(a) 11 (B) 13 (C) 16 (D) 17

Sol:



$p = 13$ $q = 17$:

↳ $n = p \times q = 13 \times 17 = 221$

↳ $\phi(n) = (p-1)(q-1) = 12 \times 16 = 192$

↳ $e = 35$ $gcd(35, 192) = 1$

↳ $de \bmod \phi(n) = 1$

$d \times 35 \bmod 192 = 1$

↳ put d = 11, 13, 16 or 17 and see who statisfies the equation (Hit & Trial).

Date :

☆ Authentication

Authentication refers to identifying each user of the system and associating the executing program with those user.

Password : The most common approach to authenticating a user identity is the use of passwords.

OTP : To avoid the problem of password Sniffing and shoulder surfing, a system can use a set of paired password.

Biometrics : Palm-or-hand reader are commonly used to secure physical access.

Algorithm Component :

- A Set K of Keys.
- A Set M of messages.
- A Set A of authenticators. :
- A function $S: K \rightarrow (M \rightarrow A)$: $S(K)$ is a function for generating authenticators from message.
- A function $V: K \rightarrow (M \times A \rightarrow \{true, false\})$: $V(K)$ is a function for verifying authenticators on messages.

* Authentication — Hash Function

Hash function, H(m) generates a small fixed-size block of data known as hash value from any given input data.

Popular hash functions are MD5, which generates 128-bit message digest or hash value and SHA-1, which generates a 160-bit digest.

Date :

* MAC: Message-authentication code

A MAC uses symmetric encryption and decryption of the message digest, which means that anyone capable of verifying an incoming message could also generate a new message.

* Digital Signature

An asymmetric approach is the digital-signature algorithm, which produces authenticators called digital signature.

* Key distribution:

Key distribution with symmetric cryptography is a major problem, because all keys must be kept secret and can't be transmitted over unsecure channel. One option is to send them out-of-band.

Asymmetric keys can proliferate – stored on key-ring.

* Implementing Security Defence

1. Security Policy: A security policy should be well thought-out, agreed upon, and contained in a living document that everyone adheres to and updated as need.

2. Vulnerability Assessment: Periodically examine the system to detect vulnerability.
   - Port Scanning
   - Check for bad password.
   - New unauthorized account.

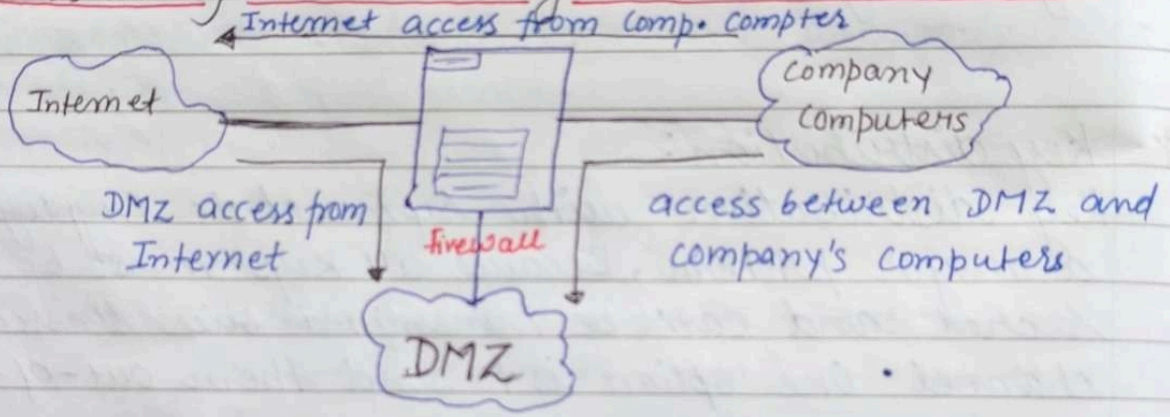
Date :

3. Intrusion Detection: Intrusion detection attempts to detect attacks both successful and unsuccessful attempts.
   ↳ Signature based detection ↳ Anomaly detection

4. Virus Protection: Modern anti-virus program are signature-based detection system, which also have the ability of disinfecting the affected files.

5. Auditing, Accounting and Logging.

☆ Firewalling to Protect Systems and Networks:



↳ Firewalls are devices that sit on the border between two security domains, restricting the traffic that can pass between them based on certain criteria.
Personal firewall, Application Proxy firewall, System-call fw.

* Computer Security classification:

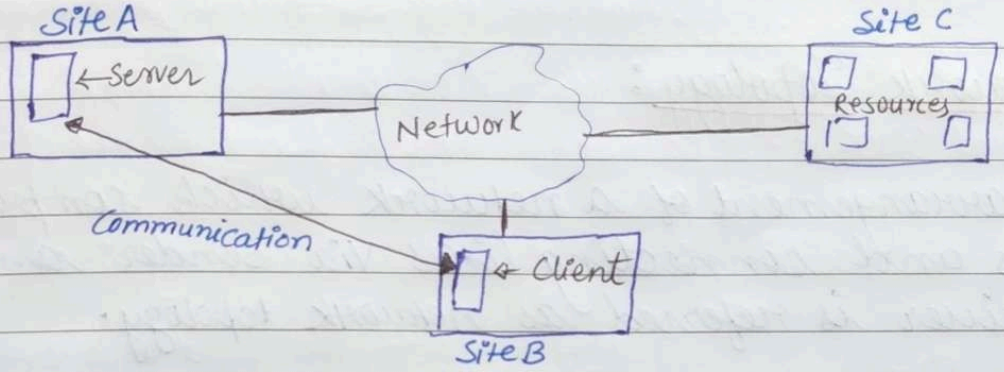
Four Division of Computer Security: A, B, C & D.
D: Minimal Security.
C: Provides discretionary protection through auditing.
B: All properties of C, each object may have Unique Sensitivity
A: Uses formal design and verification techniques to ensure Security.

Date :

☆ Distributed System:

Distributed System is collection of loosely coupled processors interconnected by a communication Network.

There are four main reason for building distributed System: Resource Sharing, Computation Speedup, Reliability and communication.



☆ Network Operating System (NOS)

A network operating system is a computer OS that is designed primarily to support workstation, PCs and older terminals that are connected on a LAN.

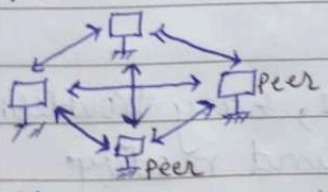
↳ It allows multiple computer to connect to share data, files, etc.

↳ Provide Security features - authentication, access control.

Services of NOS :- i) Remote login ii) Remote File Transfer

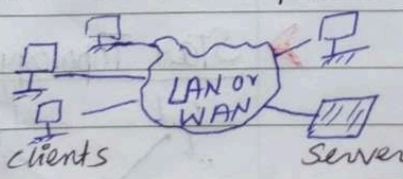
* The are two types (Implementation) of NOS

1) Peer-to-Peer



It allows user to Share N/w resources saved in a common, accessible n/w location.

2) Client-Server System



It provide user with access to resources through a Server.

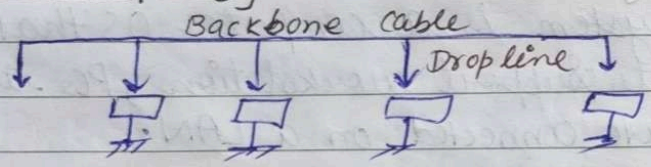
Date :

Peer-to-Peer	Client-Server System
1) Every node acts as a client and server.	→ Specific clients are connected to server.
2) Focuses on connectivity	→ Sharing the information.
3) Each peer has its own data.	The data is stored in a centralized server.
4) Less Expensive	More Expensive

★ Network Topology:

The arrangement of a network which comprise of nodes and connecting lines via Sender and receiver is referred as network topology.

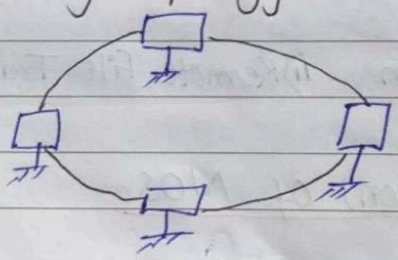
➤ Bus Topology:



Adv: It is cost effective.
- Used in small N/w.

DisAdv: Cable failure then whole n/w failure.

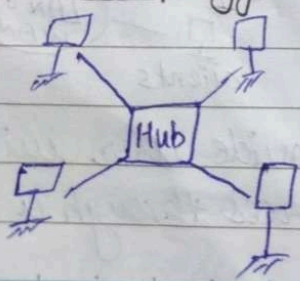
➤ Ring Topology:



Adv: Cheap to install and expand.
- Transmitting n/w is not affected by high traffic.

DisAdv: Troubleshooting is difficult.
- Adding & deleting the computer disturb network activity.

➤ Star Topology:



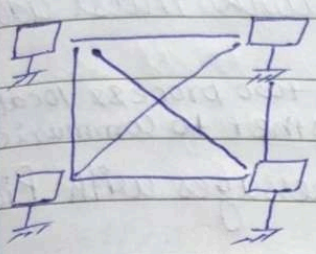
Adv: Fast performance, Easy troubleshooting.
- Easy to Setup and modify.

Disadv: Cost of installation is high.
- Hub fails then whole N/w Stopped.

Date :

1) Partial Mesh Topology.

➤ Mesh Topology: 2) Fully Mesh Topology.



Adv: Fully connected, Robust.
Provide Security and privacy.

Dis: Expensive, Cabling cost is more.
• Installation and Configuration is difficult.

➤ Tree Topology and Hybrid Topology.

★ Network Types :

	LAN	MAN	WAN
1)	Local Area Network	Metropolitan Area N/w	Wide Area Network.
2)	Connect small group of computer in an area.	Cover large regions- towns, cities, etc.	Connect various countries.
3)	Very easy to design and maintain.	Difficult to design and maintain.	Very difficult !! ...
4)	Very high Internet speed.	moderate Internet speed.	low Internet speed.
5)	Congestion in N/w is very low.	It exhibits higher N/w congestion.	It exhibits higher N/w congestion.

Date :

Communication Structure

The design of communication network must address four basic issues:

1) **Naming and name resolution** : How do two process locate each other to communicate?
   - Name Systems in the Network, Add. messages with Pid.
   - Domain name Service (DNS).

2) **Routing Strategies** : How are msg. Sent through the network?
   - i) **Fixed routing** : A path from A to B is specified in advance.
   - ii) **Virtual Circuit** : A path from A to B is fixed for the duration of one session.
   - iii) **Dynamic routing** : The path is choosen only when Msg is Sent.

3) **Connection Strategies** : How two processes send a sequence of messages?
   - i) **Circuit Switching** : A permanent link is established for the duration of the communication.
   - ii) **Message Switching** : A temporary link is established.....
   - iii) **Packet Switching** : Messages of variable length are divided into fixed length packets which are sent to destination.

4) **Contention** : The N/w is a shared resource, so how do we resolve conflicting demand for its use?

Sites may want to transmit info. over a link Simultan. Techinques to avoid repeated collision —

- i) **CSMA/CD** : Carrier Sense with multiple access / Collision Detection.
- ii) **Token Passing** : A unique message type, known as token, circulates in the System (usually ring structure).
- iii) **Message Slots** : A number of fixed-length message slots continuosly circulates in the system.

Date :

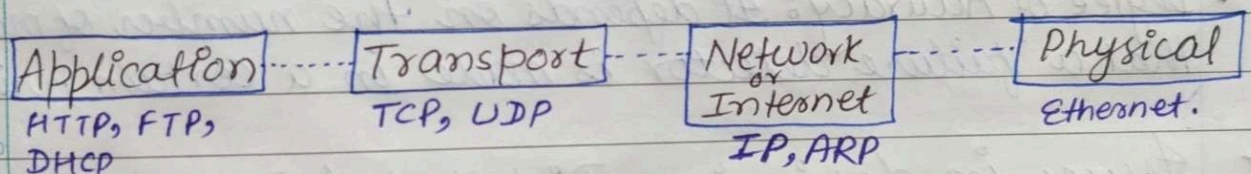
☆ Communication Protocol :

* OSI Model

Open System Interconnection

7. Application Layer : Interect directly with the user.
6. Presentation layer : Translate the data into required formate.
5. Session layer : Implement Session, maintenance of Session.
4. Transport layer : Flow & Error Control, low level network access and for Msg. transfer between clients.
3. Network Layer : Provide connection and route packets, Logical Addressing.
2. Data-link layer : Handles the frames, node-to-node Msg. delivary.
1. Physical Layer : handles the mechanical and electrical details of physical transmission of bit stream.

* TCP/IP Protocol layers :



* OSI	TCP/IP
• Open System Interconnection.	• Transmission Control Protocol / Internet protocol.
• It has 7 Layers.	• It has 4 layers.
• Follow Vertical approach.	• Follow horizontal approach.
• Transport Layer gurantees packet delivery.	• Doesn't not gurantee. But TCP/IP model is more reliable.
• Less reliable than TCP/IP.	• More reliable than OSI model.
• It is low in usage.	• It is mostly used.

Date :

★ Failure Detection:

In a distributed computing system, a failure detector is a computer application that is responsible for the detection of node failure or crashes.

↳ To detect link failure, Handshaking Protocol can be used.

Classification:

Accuracy / Completeness	Strong perpetual Accuracy	Weak perpetual Accuracy	Strong Eventual Accuracy	Weak Eventual Accuracy
Strong Completeness	P Perfect failure Detector	S Strong failure Detector.	◆P Eventually Perfect failure Dector.	◆S Eventually S.....
Weak Completeness	Q Quasi-perfect FD.	W Weak FD	◆Q Eventually Q.....	◆W Eventually .....

8-Types of failure Detector

- Degree of Completeness: It depends on the number of crashed process is suspected by a failure detector in a certain period.
- Degree of Accuracy: It depends on the number of mistake that a failure detector is made in a certain period.

★ Issues in designing Distributed System:

1. Heterogenity: The internet enables user to access services and run application over hetrogenous collection of Networks.
2. Tranparency: Locational, Migration, Replication transparency and Concurrency.
3. Fault tolerance: The distributed system should continue to function in the face of failure.
4. Scalability: As demand increases, system should accept the addition of new resources.
5. Clusters: a collection of semi-autonomus machine.
6. Openness, 7. Quality of Services 8. Reliability
9. Performance.