OPERATING SYSTEM

1) What is an operating System ? Explain Kernel.

Operating System :-

- The operating system is a program that runs all the time, as long as the computer is operational and Exit only when the computer is shuts down.

- Power-on routine-
  - When you switch on the computer what all happens before you start operating system on it.

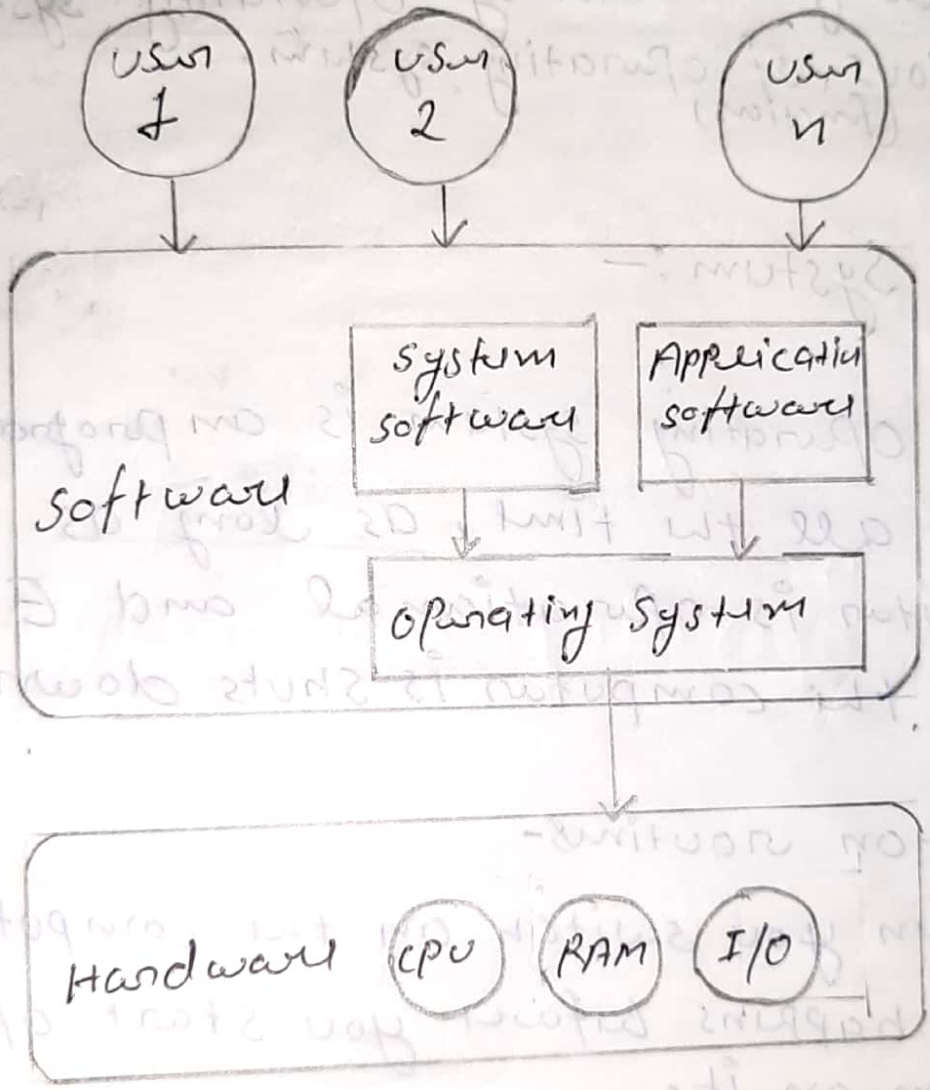
    Some information appears on the screen. this is followed by memory counting Activity.
  - Keyboard, disk drives, printer and other similar devices are verified for Proper operation.

  These Activities always occur (find) whenever the computer is switched on or reset.

  These Activities are called power-on routines.

Definition-

- An operating system is a program that performs as an interface b/w the user and computer hardware and controls the all kinds of programs.



- some popular operating systems include .linux, .windows, .Dos and .macintosh operating system.

⇒ Function of operating system

some important function of an operating system—

1. process management
2. memory management
3. File management (Authorized file) data etc)
4. security
5. command Interpretation → (user Request to send)
6. Resource Allocation. → (multiple user and more jobs Running at the same time)

(1) process management ⇒

In this, Operating system takes care of the Creation, deletion of processes and scheduling of various system resources to the different process as per their request.

(2) memory management ⇒

It takes care about allocation, deallocation and reallocation of memory space for various process, which are running at a time in computer.

(3) File management ⇒

This module takes care about all Activities related to file for data storage, retrieval, sharing, naming and protection for particular user.

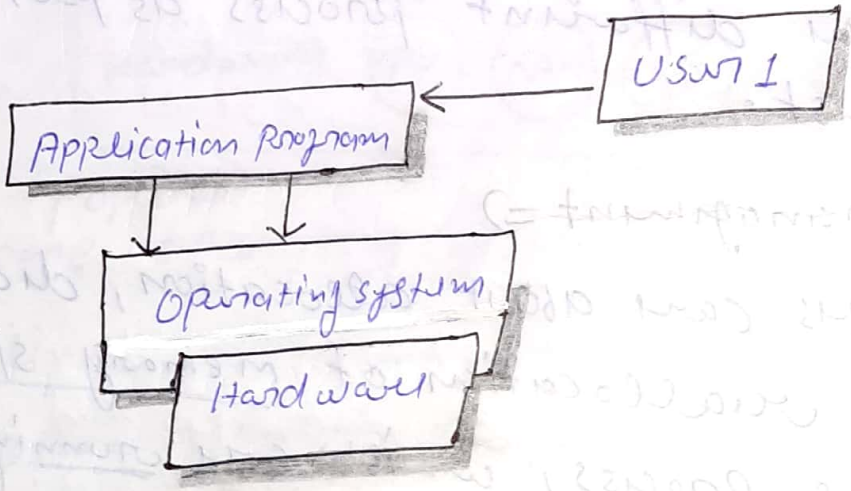
⟹ Type of Operating System

(1.) Single-User → Single Processing System :-

- It has a single processor, runs a single program and interacts with a single user at a time.
- The operating system for this system is very simple to design and Implement.

However,

The CPU is not utilized to it's full potential, because (Cond.) it sits Idle for most of the time.



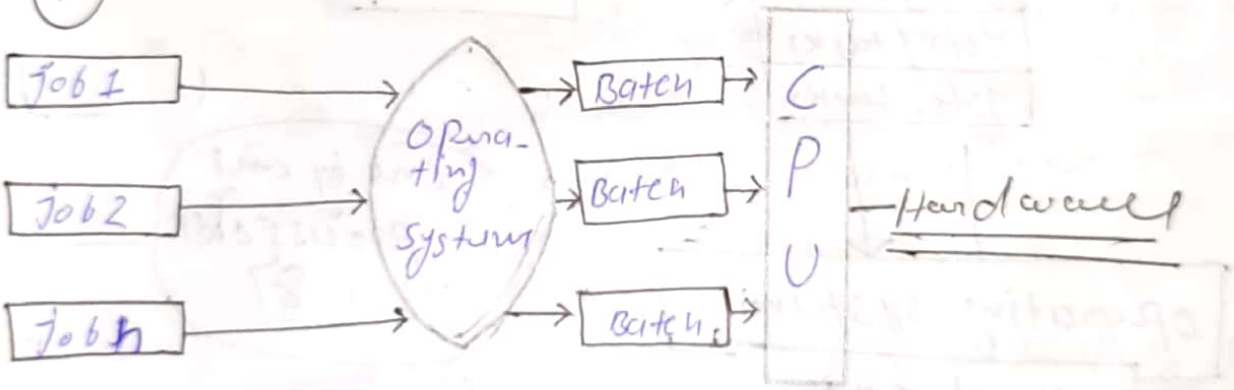
Single user - Single processor System

② Batch Operating System (Batch processing Sys.):

- This type of operating system do not Interact with the computer in directly way.
- There is an operator which takes similar jobs having <u>same</u> requirement and group them into batches.

⇒ It is the Responsibility of operator to sort the jobs with similar reqds.

(1)



Batch processing system

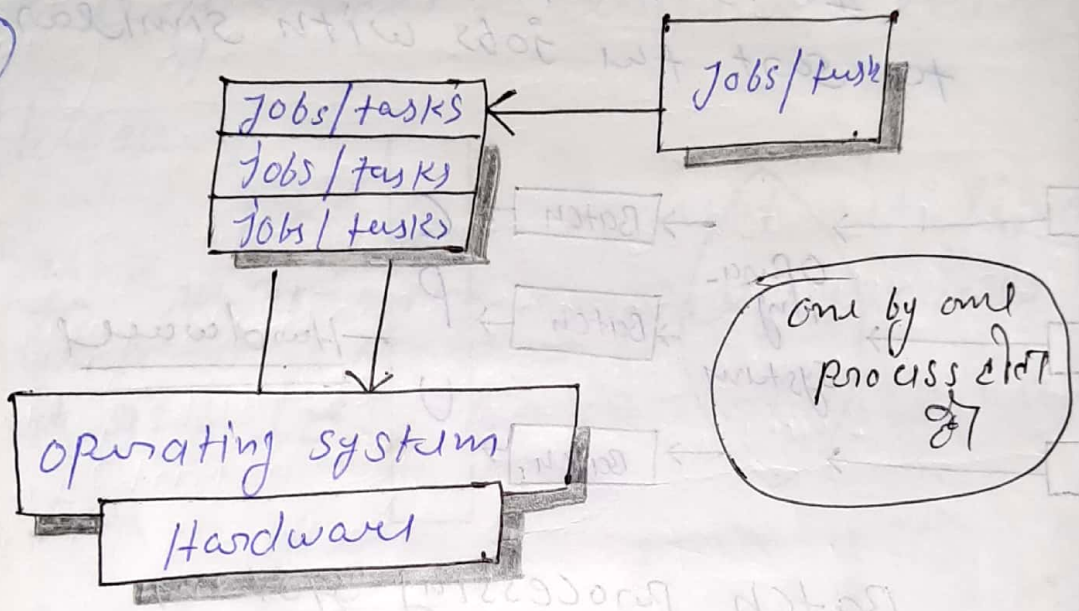
Another term ⇒

- It is commonly used in a batch processing system is job scheduling.
- job scheduling is the process of sequencing jobs so that they can be Executed on the Processor
- It recognizes different jobs on the basis of First-come-first-served (FCFS) manner.
- The batch monitor always starts the next job in the batch.

However, In Exceptional cases, you could also Arrange the different jobs in the batch depending on the priority of each batch.

- So, the batch having the highest priority could be made to run earlier than other batches.

(2)



Batch Operating System

Key point

- The batch operating system tries to reduce the CPU idle time through Operator interaction. However, it can't reduce the idle time due to I/O operations.

③ Multiprogramming Operating System :-

- The main objective of a multiprogramming operating system is to increase the system utilization efficiency.

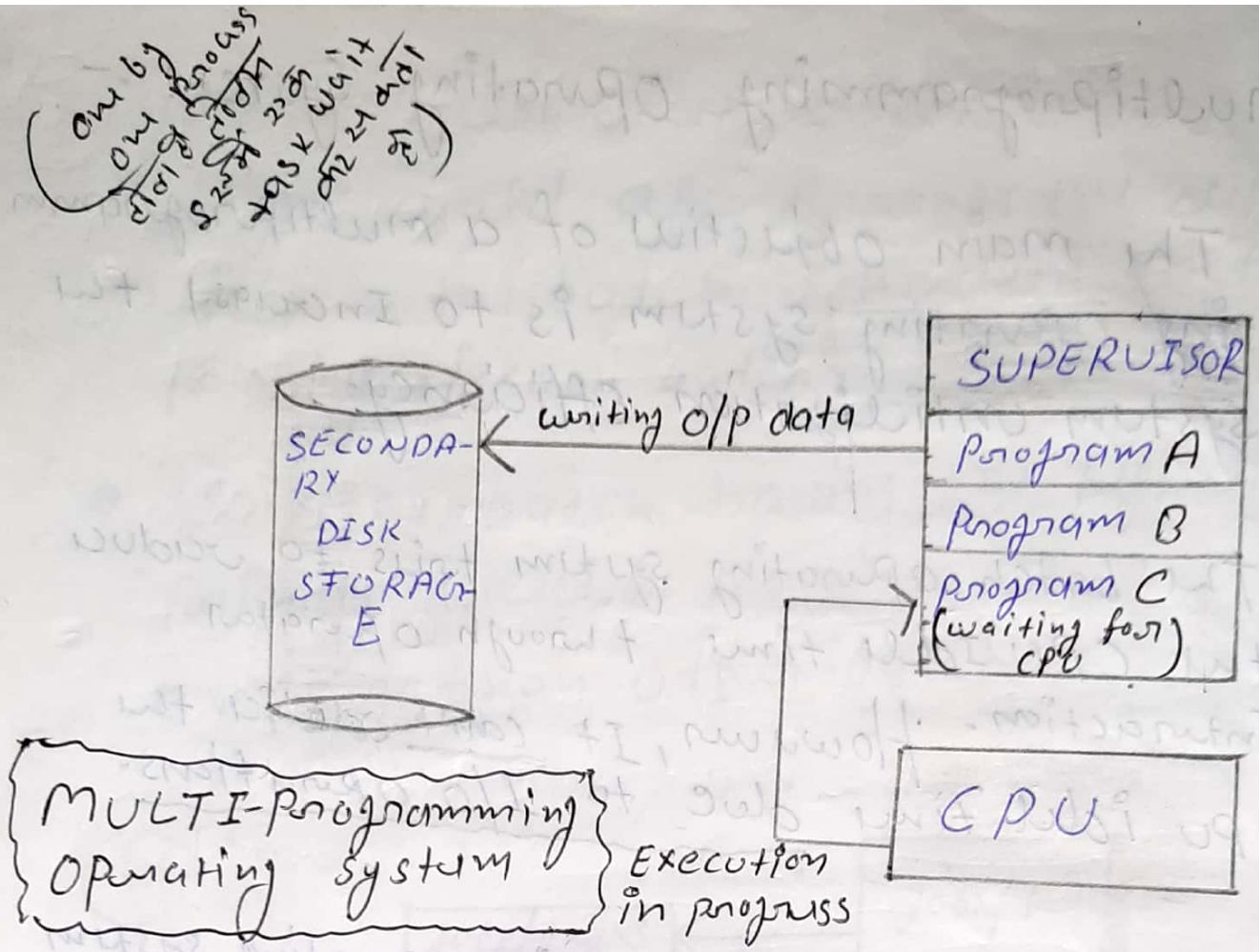
- The batch operating system tries to reduce the CPU idle time through operator interaction. However, it can't reduce the CPU idle time due to I/O operations.

- Thus, The multiprogramming operating system tries to eliminate (remove) such idle time by providing multiple computational tasks for the CPU to perform.

  - It could achieve by keeping multiple jobs in the main store.

  However, If it was not so, then it could lead to some erroneous situations leading to some time-dependent errors.

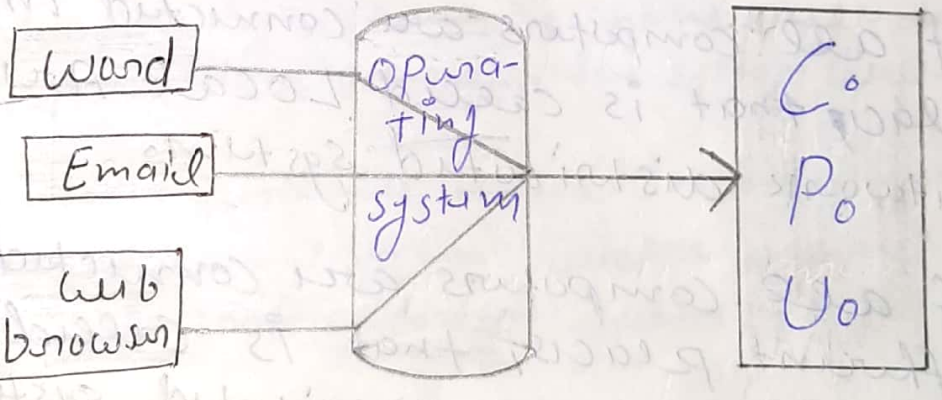
Ex - UNIX, VMS, and Windows-NT etc.



- A multiprogramming supervisor has a very difficult job of managing all the activities that take place simultaneously in the system.

(4) Time Sharing / multitasking system / multiprogramming with Round-Robin / fair share ⟹

- multitasking is a multiprogramming with time sharing.
- only one CPU but switches b/w processes so quickly that it gives an illusion (not real) that all execting at same time.
- The ~~task~~ in multitasking may refer to multiple jobs of the same program.
- main Idea is better response time and Exactly multiple process together.



Limitation ⟹
- Each tasks gets an Equal opportunity.
- Less Chances of duplicating of Software.
- CPU Idle time can be reduced.
- Reliability problems.
- Data Communication problems.

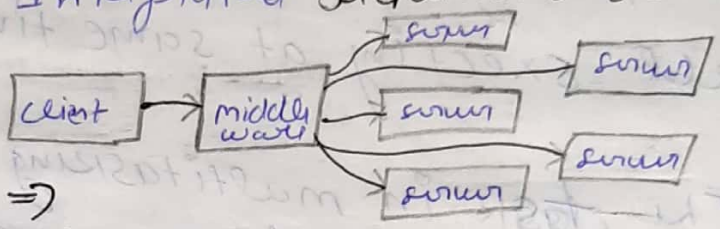
Example ⟹ multics, unix etc.

(5) Distributed Systems: ⇒

(2) middleware ⇒

- A distributed system is a network that consist of autonomous (Independent) computers that are connected using a distribution middleware.

- They help in sharing different resources and capabilities to provide users with a single and Integrated coherent (Coherent) network.

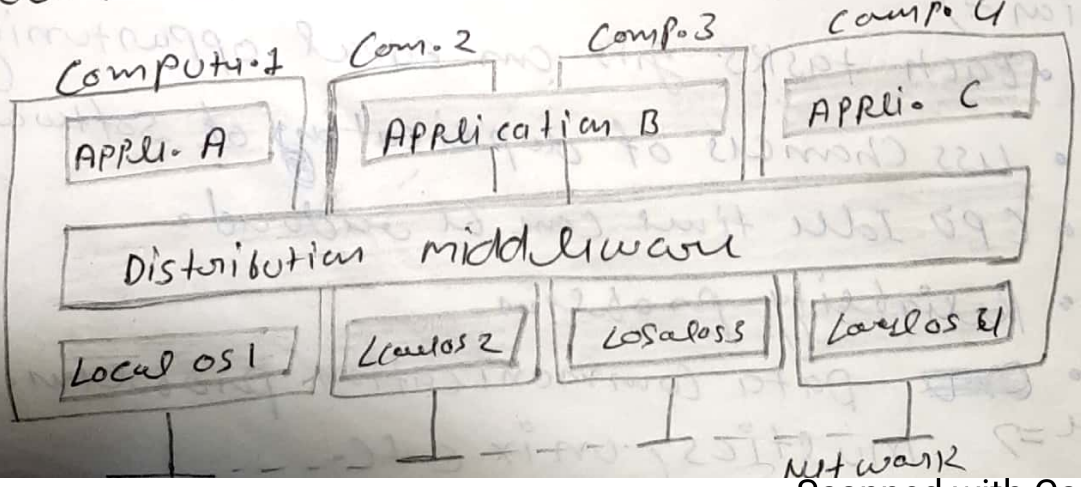


In other word ⇒ (1) Introduction ⇒

- When using several Independent computers and it is used as a single system. that is called Distributed system.

- If all computers are connected in one place, that is called Local Area Network distributed system.

- If all computers are connected in different places, that is called a Wide Area Network Distributed system.



(6) Real-time operating system :⇒

- A realtime system is defined as a data processing system in which the time interval required to process and response to Inputs is so small that it controls the Environment.

Respond quickly

एक Input को Process करने में लगने वाला time कम होना चाहिए।

एक ऐसा task जिसका Input करने पर कुछ ही seconds के अंदर हमें output देने मिल जाए उसे कहते हैं Real time system - Like → camera, scanning, etc.

- Time taken from Input to output task is Response time. (quickly).
- It responds to Inputs immediately (real-time).
- How the task is completed within a specified time delay.
- The operating system has to respond quickly.

⇒ A Real-time operating system most have well-defined, fixed time constraints, otherwise the system will-fail.

Ex - scientific Experiments, medical Imaging system, Industrial control system, weapon system, Robots, air traffic control system etc.

There are two types of RTOS

- Hard Real time system
- Soft Real time system

① Hard Rial Time system :-

- A Hard Rial time system guarantees that critical tasks complete on time.
- In this secondary storage is limited or missing and the data is stored in ROM.
- In this systems, virtual memory is almost never found.

② Soft Rial time system :—

- A critical Rial-time task over other tasks and retains the priority until it complete. (इस system में critical tasks के आगे में कोई भी task आ नहीं होता है, कितने इसमें पहले वही task complete होता है जो प्राथमिकता critical होता है)
- These systems have limited utility than Hard Rial-time system.

Advantages =>

① maximum consumption =>

Gives maximum output, while using all the resources and keeping all device Active.

② Task switching =>

Less time is needed in shifting the task which is around 3 micro seconds.

③ FOCUS on Application

④ used in embedded system also

⑤ Error free.

⑥ 24-7 system.

⑦ Best memory Allocation.

(1) Logical vs physical Address in operating system [memory man.]

Logical Address ⇒

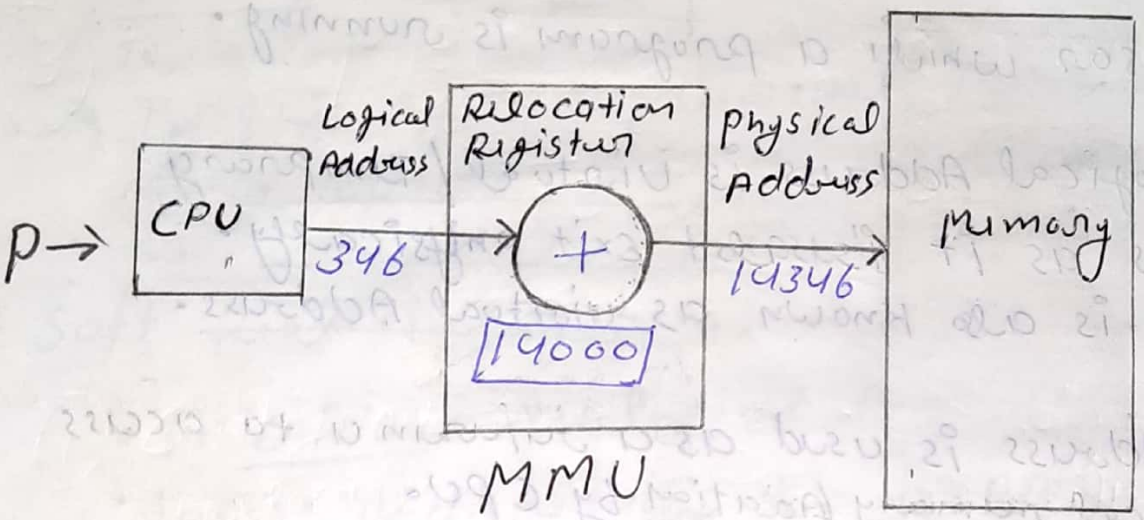
- Logical Address is generated by CPU or Processor while a program is running.
- The logical Address is virtual / Temprary address as it does not exit physically. Thus it is also known as virtual Address.
- This Address is used as a reference to access the physical memory location by CPU.
- Logical Address space is used for the set of all logical addresses generated by programs perspective.

Physical Address ⇒

- Physical Address identifies a physical location of required data in a memory.
- The user never directly deals with the physical Address but can access by it's corresponding logical Address.
- The User program generates the logical Address and thinks that the program is running on logical Address but the program needs physical memory for it's Execution.
- Thus, Logical Address must be mapped to the Physical Address by MMU (memory management unit).

before they are used.

- Physical Address space is used for all physical addresses corresponding to the logical addresses in a logical address space.

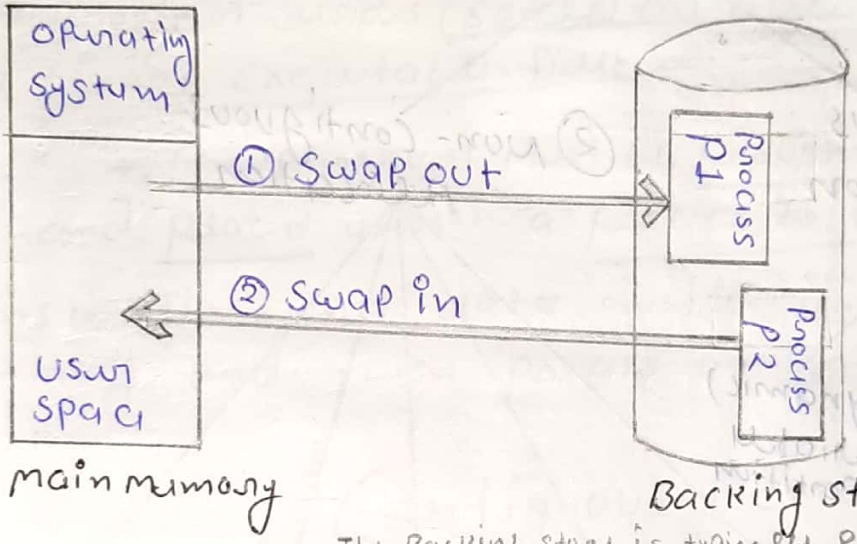


Memory Management Unit

Note→ The run-time mapping from virtual to physical address is done by a hardware device called a memory management unit. (device)

(2) SWAPPING - MEMORY MANAGEMENT

- A Process Nuds to bu in thu mumory to bu Executud.
- A Process can bu swappud temparanily out from main mumory to backing storu, and thun brought back into memory for continuud execution.



- Baking storu :- Thu Backing Storu is typically part of thu hard disk, that is usud by a paging or swapping system to storu information in main mumory.

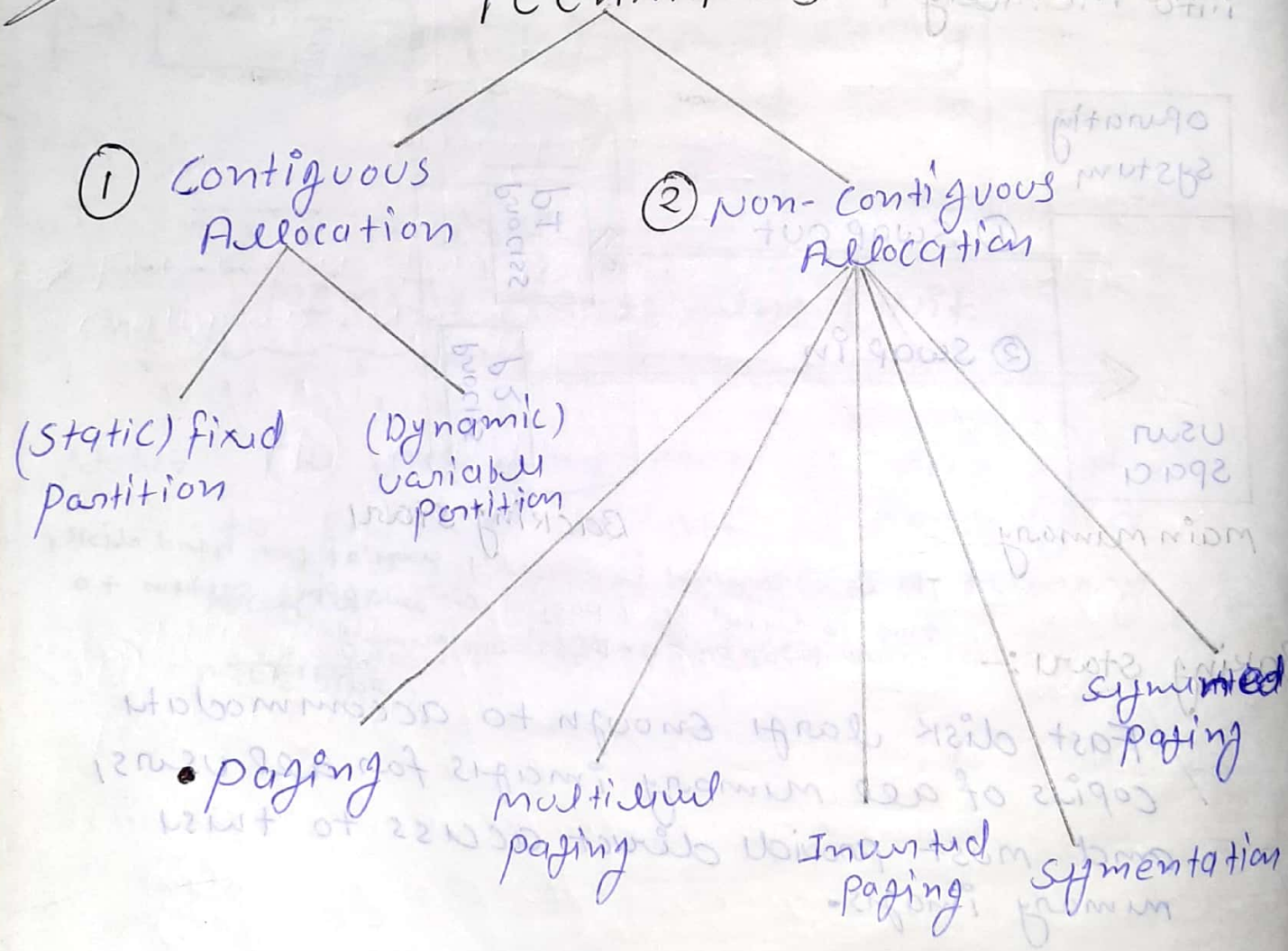
  Fast disk largu Enough to accommodatu copius of all mumory imagis for all usurs, and must provido direct accuss to thusu mumory imagis.

- Roll out, Roll in :-

  Whun a highur-priority Process Arrivus, a lowur priority Process is swappud out, and thun swappud back in whun thu highur-priority process finishus.

- Transfer time :-
  - Total transfer time is directly proportion to the amount of memory swapped.
  - Swapping requires backing store (Normally a fast disk).

(3) Memory Management Techniques



Memory management

- Memory is one of the central resource required for a computer operation.
- Memory is organized in words. Each word may consist of one byte, two byte, or four byte or more as per operating system design.
- Each memory location is addressable and unique for address.
- A program resides (settles) on a disk (memory) as a binary executable file.
  - → The program must be brought into memory and placed within a process to be executed.
- Instruction and data must be loaded into the memory and this process managed by the OS.

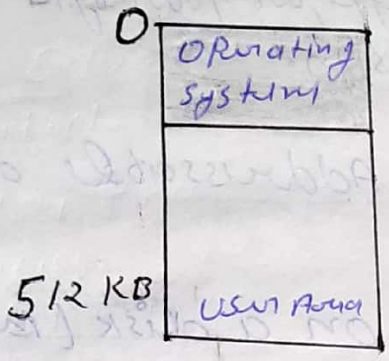
① Contiguous Memory Allocation

- Contiguous memory Allocation is a memory allocation technique.
- It allows to store the process only in a contiguous fashion.
- Thus, Entire (overall process) process has to be stored as a single (P1) Entity at one place inside the memory.

Scanned with CamScanner

Key point

- The main memory (RAM / virtual / Primary / Logical) is usually divided into two parts (Partition).
  - → first is operating system loaded into memory.
  - → second is used for user programs.



Memory partition

→ There are two popular technique used for contiguous memory Allocation.

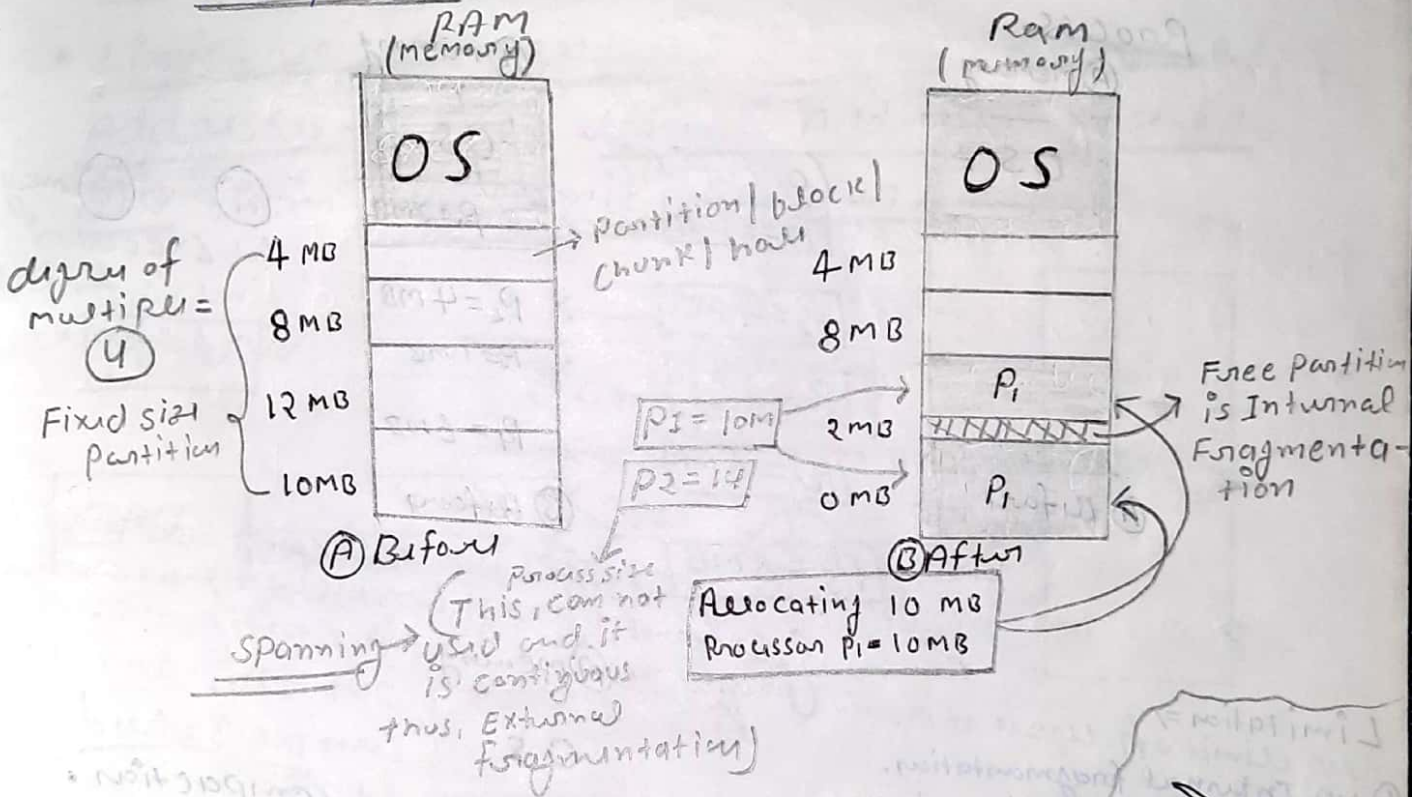
(1) Contiguous memory Allocation technique

(A) static Partitioning or fixed size partition

(B) Dynamic Partitioning or variable size partition

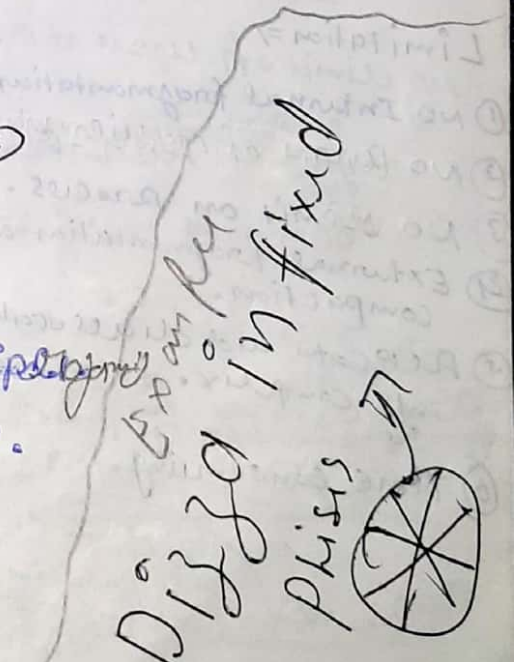
(A) Fixed Partition scheme :⇒

- Static partitioning is a fixed size partition.
- In this technique, main memory is pre-divided into fixed size partitions.
- The size of each partians are fixed and can not change.
- Each partition is allowed to store only one process. and it's size may same or not.



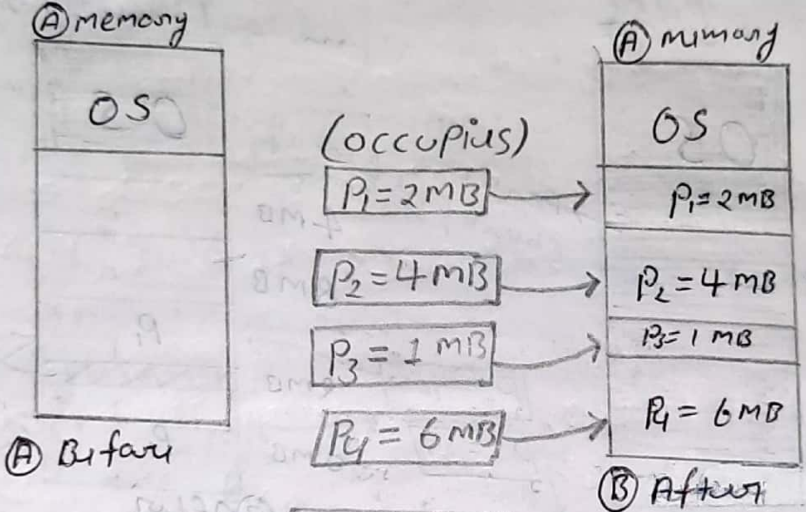
Limitation of this technique ⇒

1. Internal fragmentation
2. Limit of process size
3. Limit on degree of multiprogramming
4. Spanning is not allowed.
5. External fragmentation.



(B) Variable size Partition :⇒

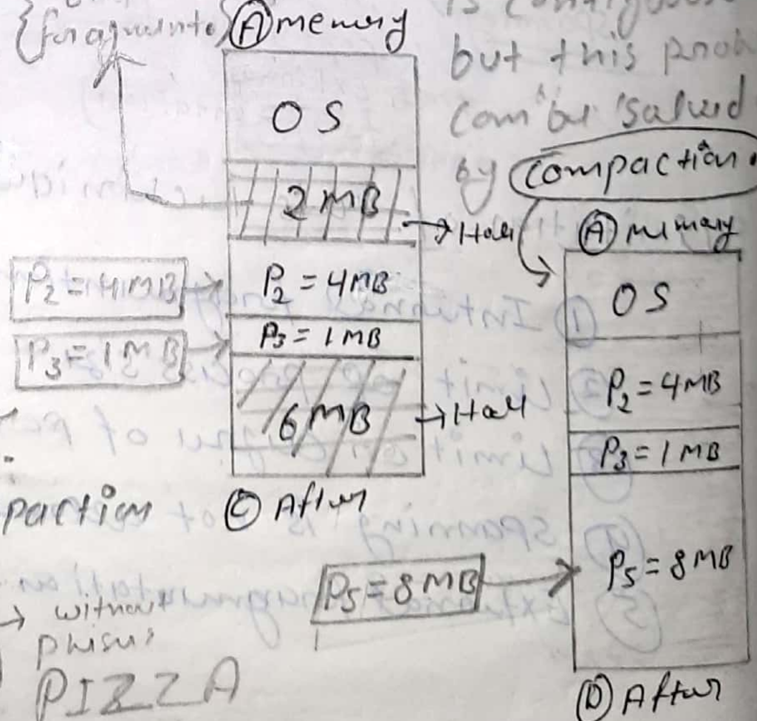
- Dynamic partitioning is a variable size partition.
- It performs the allocation Dynamically.
- When a process Arrives, A partition of size Equal to the size of process is created.
- Then, that partition is allocated to the process.



(P1) and (P4) may be executed time is finished on block.

P5 = 8MB → This process can't used in this memory because this technique is contiguous. but this problem can be solved out by compaction.

{External fragmentation}



Limitation ⇒
no limit of degree of multi.

1. No Internal fragmentation.
2. ~~No limit on process size.~~
3. No limit on process.
4. External fragmentation and compaction.
5. Allocate and deallocate is complex. as compare to the fixed partition.
6. Time consuming for compaction

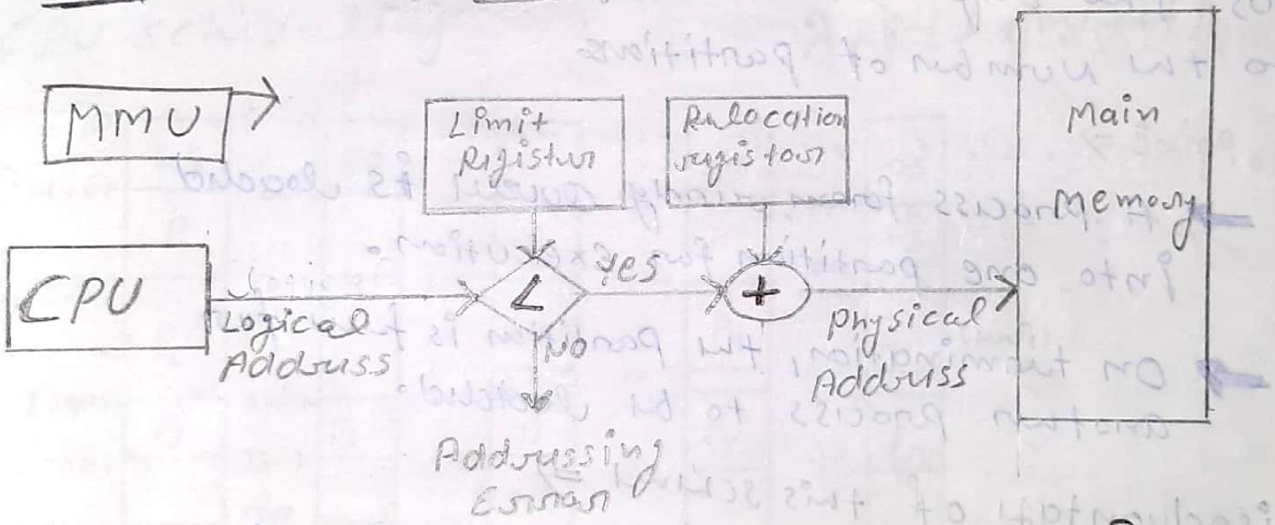
लेकिन यहां Problem finishes नहीं होती है क्योंकि यहां 3 का P. के बाद 2 P. काल बड़ा रहता है → without phase

PIZZA

Continuous Allocation

⟹ Singel-partition Allocation :⇒

- Relocation-register scheme used to protect user process from each other.
- Relocation register contains value of smallest physical address,
- Limit register contains ranges of logical addresses - each logical addresses must be less than the limit register.



- The operating system resides in the Lower Ram memory.
- The user processes Execute in the Higher memory.
- There is always possibility that user processes may try to access the lower memory either accidentally or intentionally thereby by causing loss of OS code and data.
- This protection is usually provided by the use of a limit register and relocation register.

Page-26

Multiple partition Allocation

- Multiprogramming refers that there are many processes residing in the memory.
- So that the CPU can switch (Creating and breaking) between these processes.
- The simplest way is to divide the user area into fixed number of partitions.
- Each fixed partition to hold one user process.
- Thus, The degree of multiprogramming is Equal to the number of partitions.

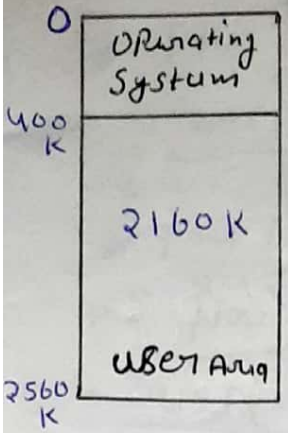
Key Points ⇒

- A process from ready queue is loaded into one partition for Execution.
- On termination (free), the partition is free for another process to be loaded.

Disadvantage of this scheme ⇒

- When partition are fixed size is the selection of partition size. If the partition size is too small then the large Programs, so can not be run.
- If partition size is also Big then programs, so can be run but space partion size is waste.

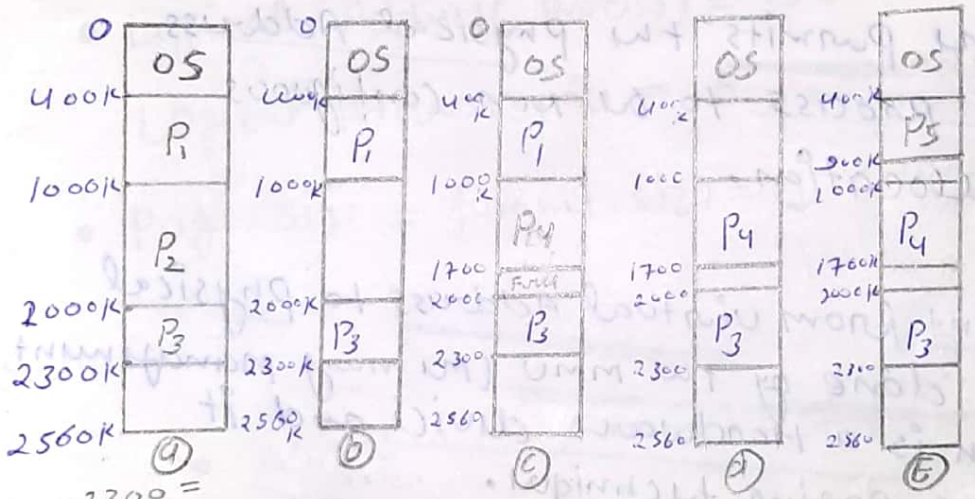
but this problem is solved out by the Variable Size partition.



Job queue		
Process	Memory	Time
P_1	600K	10
P_2	1000K	5
P_3	300K	20
P_4	700K	8
P_5	500K	15

Scheduling Example

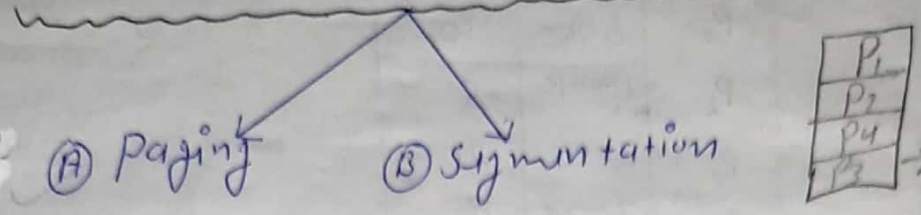
Total memory available : 2560K
Resident operating system : 400K
Memory available for user : 2560 − 400 = 2160K
Job queue : FCFS
CPU scheduling : RR (1 time unit)



2560 − 2300 = 260 ≠ P_4

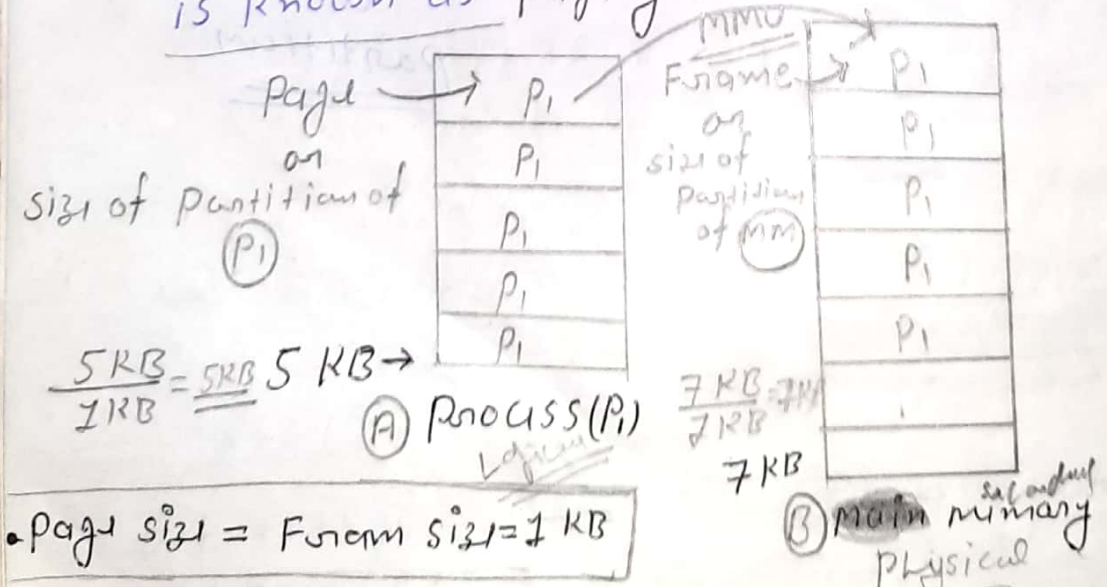
⇒ Variable partition

② Non-contiguous memory Allocation techniqu



Ⓐ Paging ⇒

- Paging is a memory managment scheme that Eliminates the need for contiguous Allocation of physical memory.
- This scheme permits the physical Address space of a process to be non-contiguous memory Allocation.
- The mapping from virtual Address to physical Address is done by the MMU (memory managment unit) which is a Hardware device and it is known as paging technique.



- Page size = Frame size = 1 KB

- The physical Address space is conceptually divided into a number of fixed-size blocks, called frames.
- The Logical Address space is also splitted into fixed-size blocks, called pages.
- The page size is always equal to frame size and viseversa. [page size = Frame size]

Example ⇒

- physical Address (M|m) = 12 bits, then
  physical Address space = 4 K words
- Logical Address (Process) = 13 bits, then
  Logical Address space = 8 K words
- page size = frame size = 1 K words [Assumption]

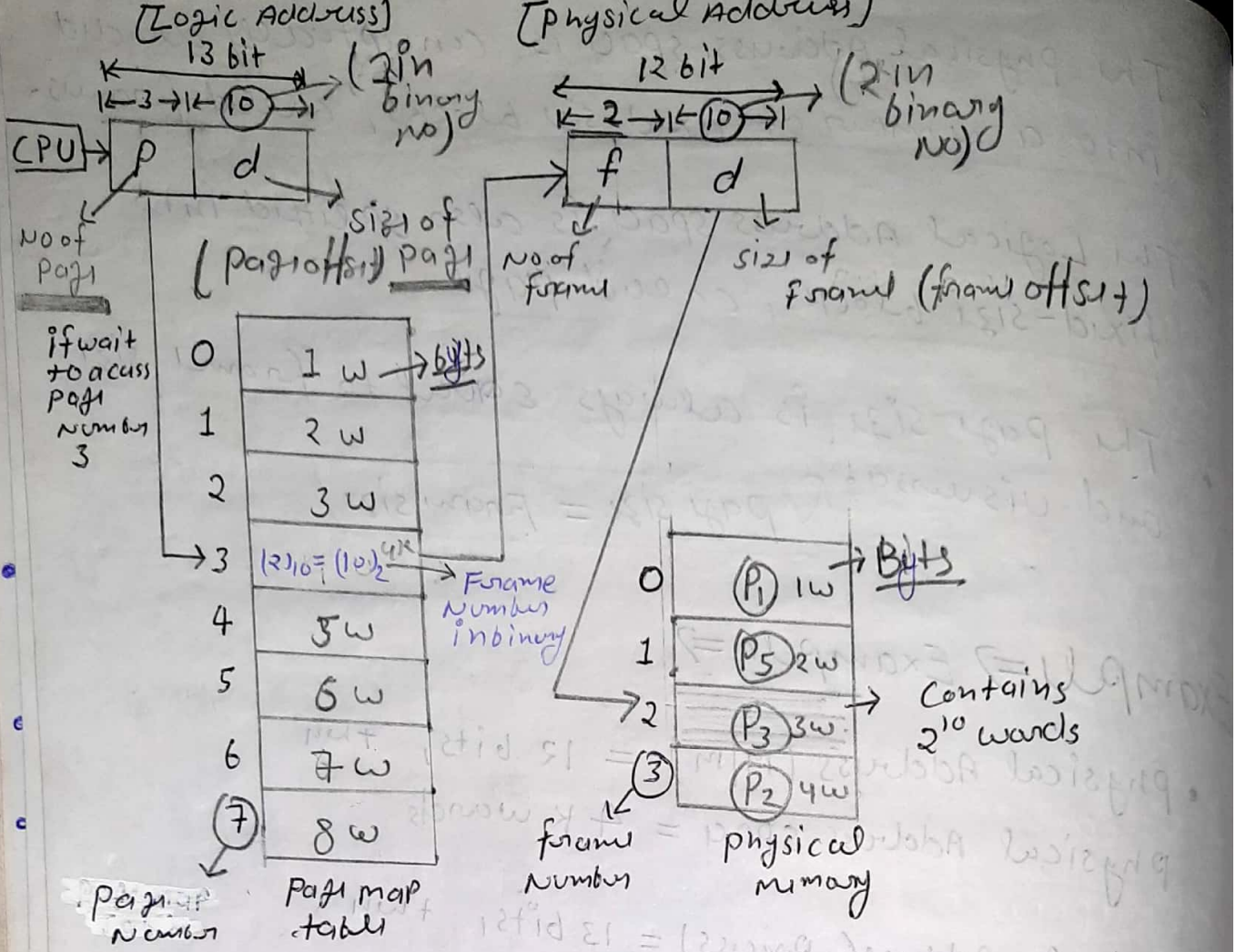
Solution ⇒

- Number of Frames = $\frac{\text{physical Address space}}{\text{Frame size}}$

$$= \frac{4K}{1K} = 4 = (2)^2$$

- Number of page size = $\frac{\text{Logical Address space}}{\text{page size}}$

$$= \frac{8K}{1K} = 8 = (2)^3$$



TERMINOLOGY

- Logical Address or virtual Address (Represents in bits):
  - An Address generated by the CPU.
- Logical Address space or virtual Address space (Represented in words or bytes):
  - The set of all logical Addresses generated by Program.
- physical Address (Represented in bits):
  An address actually available on memory unit.
- Physical Address space (Represented in word or bytes):
  The set of all Physical addresses corresponding to logical Addresses.

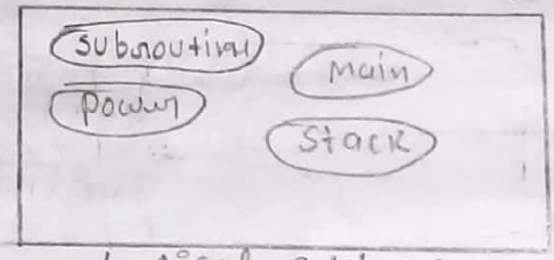
(B) SEGMENTATION

Introduction :-

- A process is divided into segments.
- The chunks that a program is divided into which are not necessarily all of the same length [limit] is called segments.
- It is also non-contiguous storage allocation.

Non-contiguously memory → paging (Fixed size, physical memory)
→ segmentation (variable size, Logical memory)

- Each segment has (a name → segment number) and (a length → size (Limit)) which is loaded into physical memory.



Logical Address

Type of segmentation :-

① Virtual memory segmentation →

- Each process is divided into a number of segments, not all of which are resident at any one point in

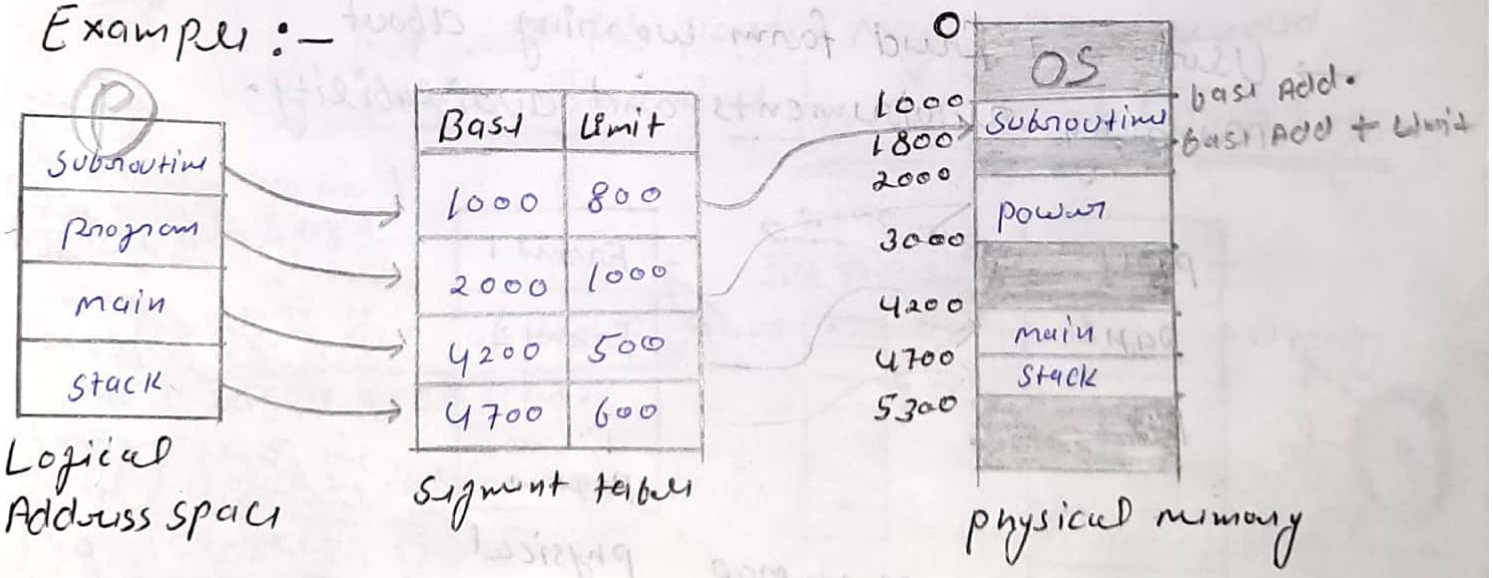
② Simple memory segmentation →

- Each process is divided into a number of segments, all of which are loaded into memory at run time though not necessarily contiguously.

Segment table :-

- A table stores the information about all such segments and it is called segment table.
- It maps Two Dimensional Logical Address into One-D physical Address.
- It's Each table has :
  - Base Address :→
    It contains the starting Physical Address where the segments reside in memory.
  - Limit :→
    It specifies the length of the segment. and Limit is Always greater than segments.

Example :-



Advantage :-

- No internal fragmentation
- Segment table consumes less space in comparison to page table.

Disadvantage :-

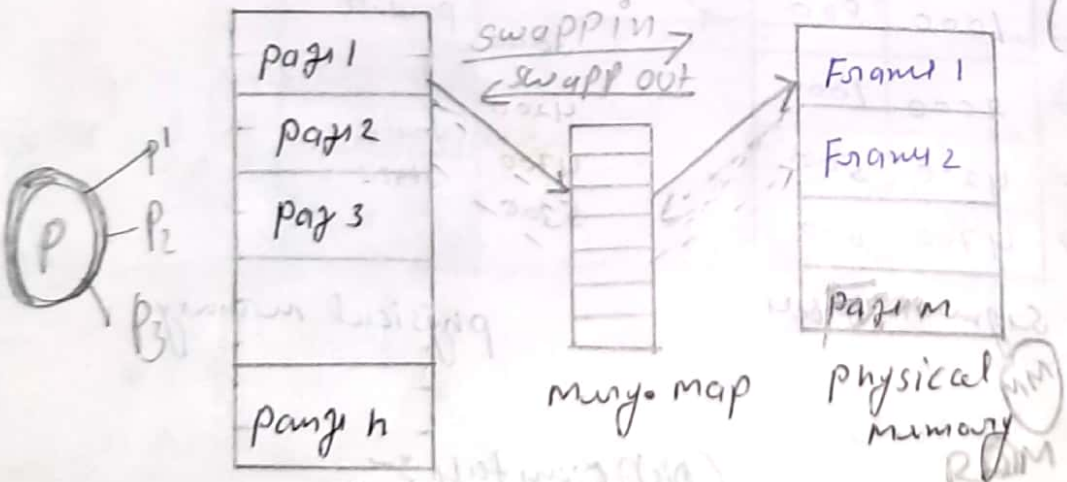
As processes are loaded and removed from the memory, The free memory space is broken into little pieces, due to External fragmentation.

④ VIRTUAL MEMORY

Introduction:-

- Virtual memory is a computer system technique that allows execution of processes that may not be entirely in memory.
- Also virtual memory allows mapping of a large virtual address space onto a smaller physical memory.
- It also raises the degree of multiprogramming and increases CPU utilization.
- Because of above features,

→ Users are freed from worrying about memory requirements and availability.



Virtual to physical memory mapping (n >> m).

Need for virtual memory technique :-

- Every process needs to be loaded into physical memory for execution.
- This Approach is to MAP the entire Logical space of the process to physical memory as in the case of paging and segmentation.

Note :→

→ Many time the entire process need not be in memory during execution. The following are some of the instances to :-

- static declaration of Arrays lists.
- Key point — Even though (भले ही) entire program is needed, but all it's parts may not be needed at the same time because of Overlays.

Advantages :-

(i) Only part of the program needs to be in memory for Execution.

(ii) Logical Address space is much larger than physical Address space.

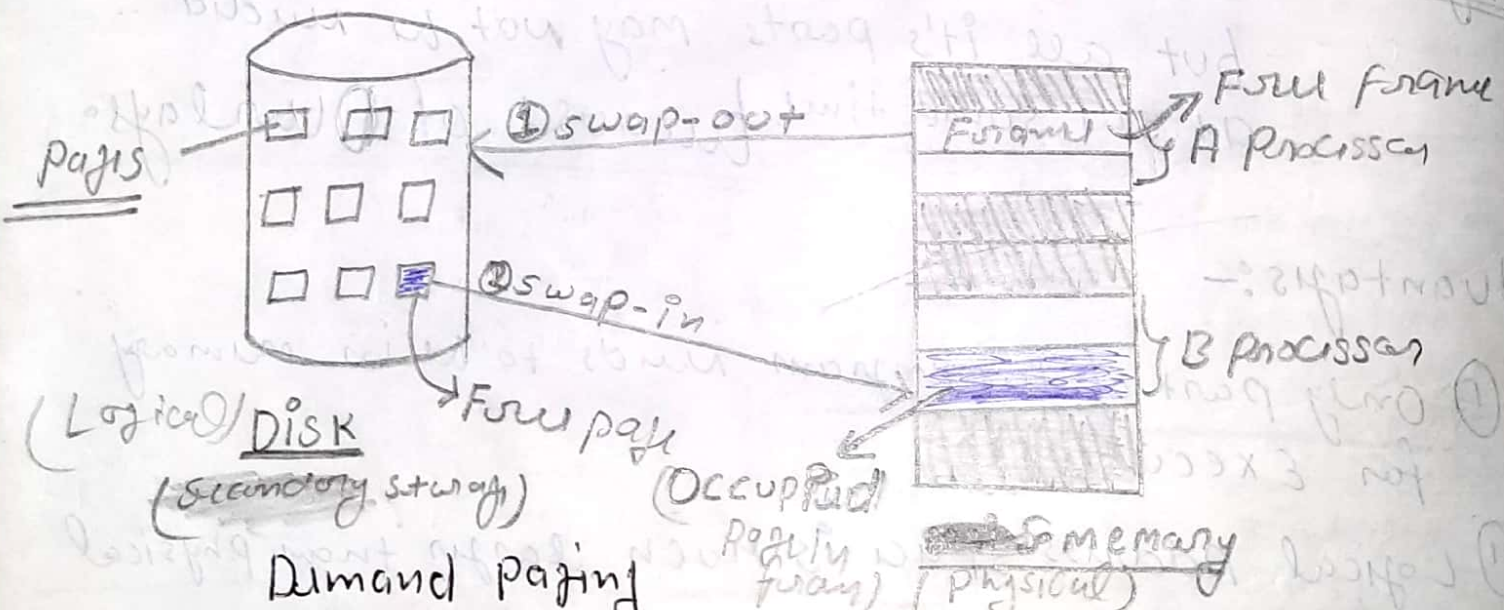
(iii) Need to Allow pages to be swapped in and swapped out.

(5) Dumand paging

Introduction:-

- In this technidal, A page is brought into the mumary for it's execution only when it is dumandid.
- this technidal is combination of paging and swapping.
- it is also know Lazy swapper bucaus swaps the pags only when it is nucid. by Program

[from secondary mumary (pags) (disk) to main mumary (Frams)]



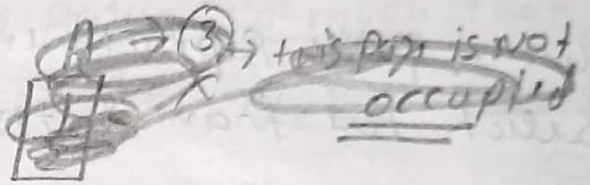
Dumand paging Diagram (Swapped from secondary storage to main mumary)

Advantages :-

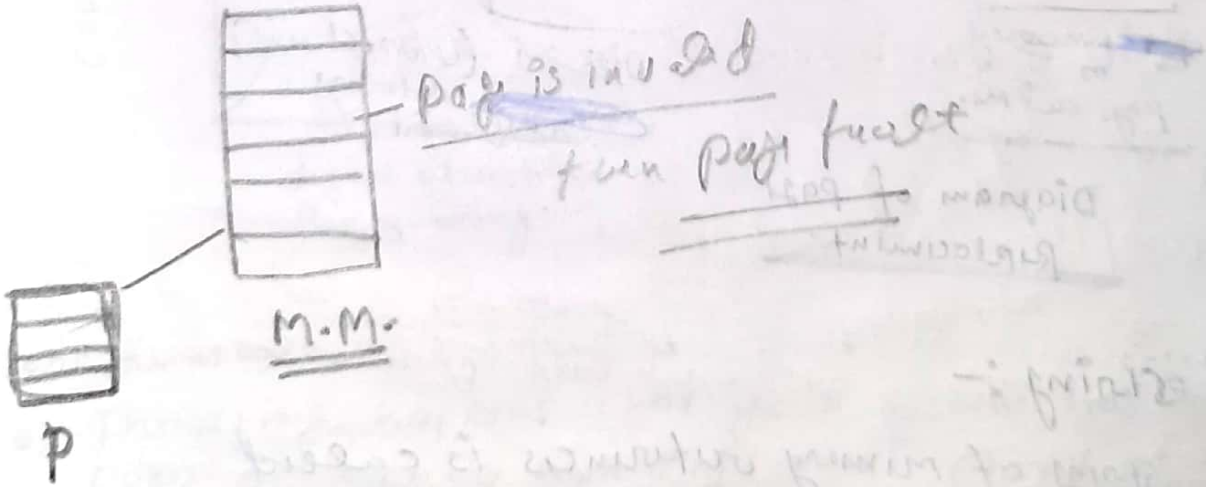
- Reduce memory requirement.
- Swap time is also reduced.
- Degree of multiprogramming. (Increases)
  - only loads pages that are demanded by the executing process.

Disadvantages :-

- page fault,



if page is in-valid then page-fault trap occurs.



⑥ Page Replacement Algorithm

- It is the technique used by operating system to decide which memory pages to swap out.
- it is also decided that in memory, How many frames to Allocate to each process.
- When page Replacement is required, we must select the frames that are to be replaced.

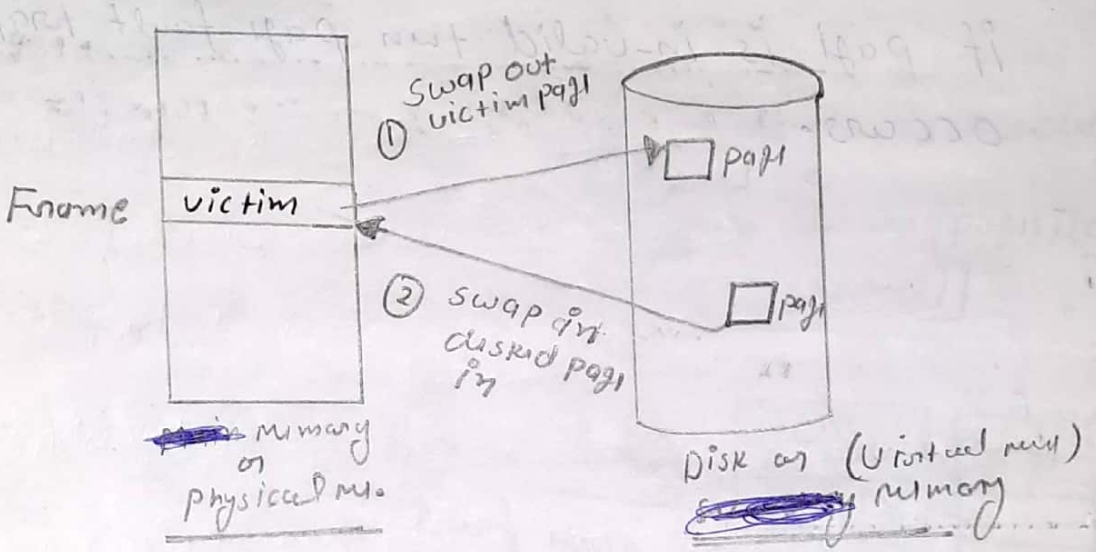


Diagram of Page Replacement

Reference string :-

String of memory references is called Reference string.

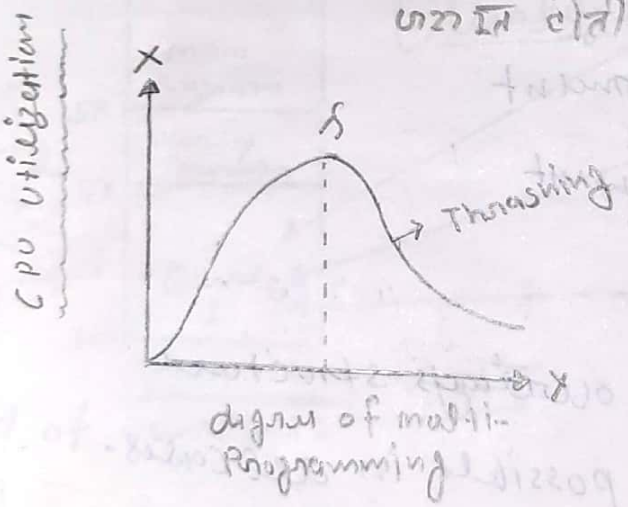
Page fault :-

A page fault happens when A running Program (Process) Accesses a memory page that is mapped into the virtual memory, but Not loaded into physical memory. (main memory)

(7) Thrashing / Disk Thrashing

Introduction:-

- Thrashing is a condition or a situation.
- Thrashing is a स्थिति steady state in which our CPU perform 'productive' work less and 'swapping' more
- CPU is busy in swapping pages, so much that इतना ज्यादा CPU can not respond to user program [process] as much जितनी ← as required. (CPU pages को swap in और swap out करने में इतना ज्यादा busy रहता है की CPU user program को उतना Respond नहीं दे पाता है जितनी उसको जरूरत होती है)



In short term ⇒

- Thrashing happens that your system has to swap pages at such a high rate that chunk of the CPU time is spent in swapping, then this state is known as Thrashing.
- So Effectively during thrashing, the CPU spends less time in productive work and more time in swapping.

To resolve thrashing you can do any of the suggestion below:-

1. Increase the Amount of RAM in the computer.
2. Decrease the Number of Programs being run on the computer.
3. Adjust the size of the swap file.

Overlays Advantages ⇒

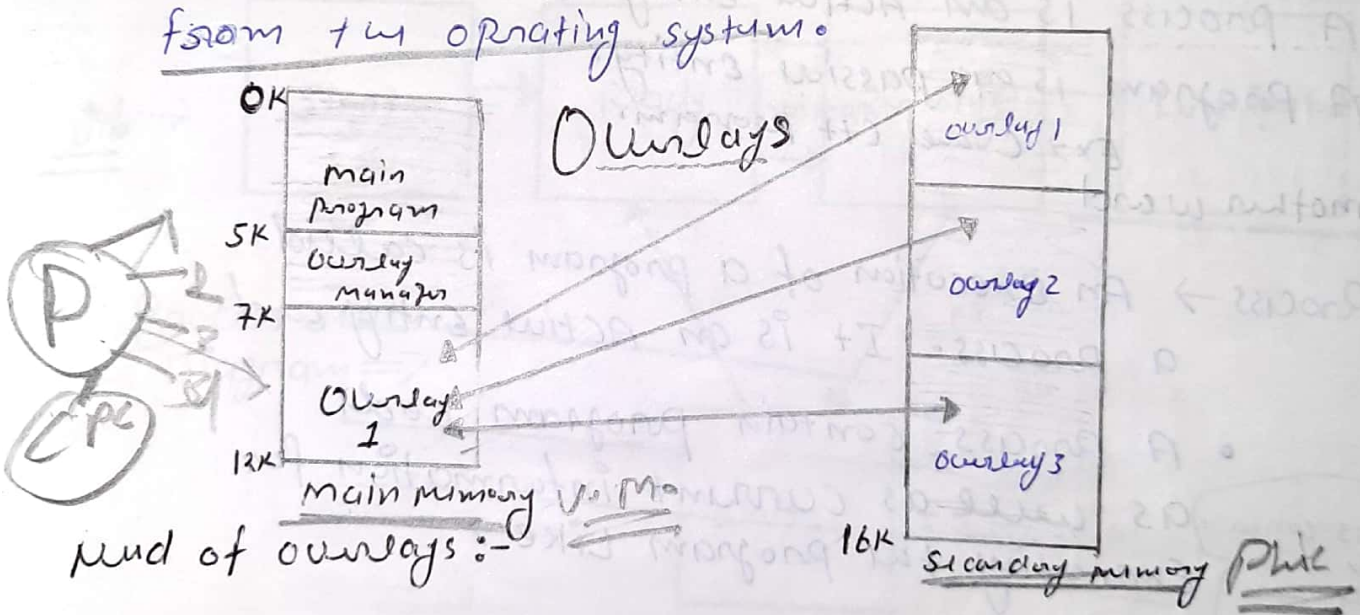
- Reduced memory requirement
- Reduced time requirement

Disadvantages ⇒

- programming design of overlays structure is complex and not possible in all cases.
- Programmer must know memory requirement.

⑧ Overlays

- Overlaying is a programming method that allows programs to be larger than the main memory (RAM).
- A overlays are a piece of code.
- The process to be executed can be divided into segments and [overlay split the program into small block]
- Overlays help swap these segments from the CPU to memory and vice versa. without support from the operating system.



Need of overlays :-

- overlay keep in memory only those instruction and data that are needed at any time.
- It is needed when process is larger than amount of memory Allocated to it.
- overlay Implemented by user, No special support needed from operating system.

Q-1

UNIT-II

- Process
- Process state
- Process control block

जरूर मांग दो

Process :→

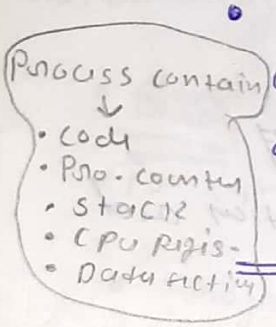
- A Process is a program in Execution.
- A Process is executed in sequentially, one instruction at a time.
- A Process is an Active entity and A Program is a passive entity. (Inactive)

Ex→ C and C++ Program.

In another word –

Process → An Execution of a program is called a process. It is an Active Entity and

- A Process contain program code as well as current information for Executing the program Like →



⇒ It has program counter which store Address of next instruction. and

⇒ It has stack, CPU Register, and Data Section.

Text section → A Process sometimes known as the Text section also include the current Activity represented by the value of Program counter.

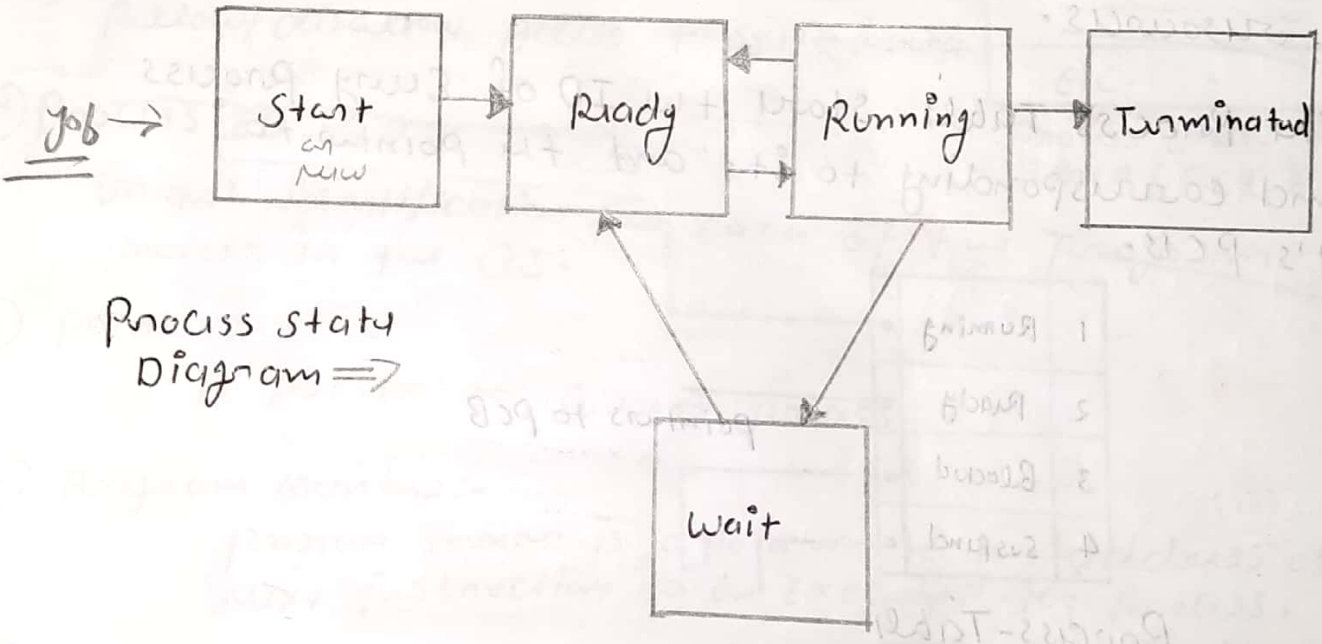
Stack → It contains the Temporary Data such as returns Addresses.

Data Section → contain global variable.

Process State :→

- The current activity of a process is known as it's state.
- When A process executes, It passes through different states.
- These states may differ in different operating systems.

In General, A process can have one of the following five states at a time.



Process state Diagram ⇒

① Start :- / New
This is initial state when a process is started / created.

② Ready :-
The process has acquired / Receive the required resource and is waiting to be Assigned to a processor.

③ Running :- Instructions are being Executed.

④ waiting :- The process is waiting for some event to occur.

⑤ Terminated :- The process has finished execution.

Process Control Block:→

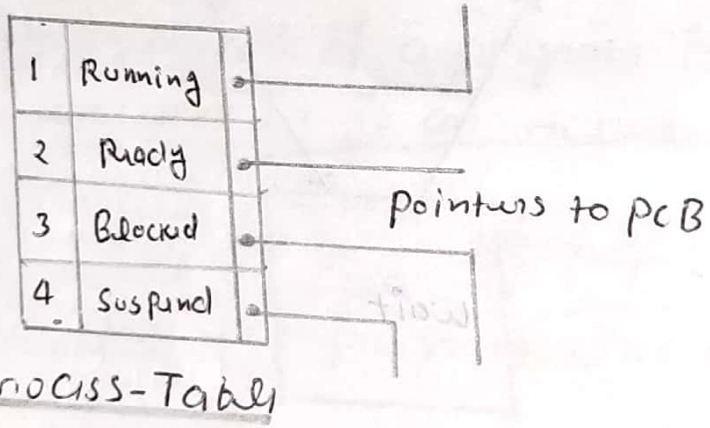
We should learn or understand about PCB PT before know about Process Table.
PCB

Process Table:-

→ The operating system manages and controls the resources by having a Table.

→ Table's are important data structures to store information about every process and resources.

→ The Process Table store the ID of every process and corresponding to it. and the pointer to it's PCB.



Process-Table

Process Control Block (PCB):-

→ A Process control Block is a data structure maintained by the Operating System for every process.

Note:→ Some operating system maintained only PCB in that PCB has all entries stored by a process Table.

→ A PCB represents a process in an operating system.
→ The PCB contains information that makes the process an active entity.

- A PCB keeps all the information needed to keep track of a process—

Process ID
state
pointers
Program counter
CPU Registers
⋮
etc.

Process control Block (PCB)

(1) Process state :-
Current state of the process.
i.e. Ready, Running, waiting etc.

(2) Process Privileges :-
Allow/disallow access to system resources.

(3) Process ID :-
Unique identification for each of the program or process in the OS.

(4) pointer :-
A pointer to parent process.

(5) Program counter :-
Program counter is a pointer to the address of next instruction to be executed for process.

(6) CPU Registers :-
Various CPU registers where process need to be stored for execution for running state.

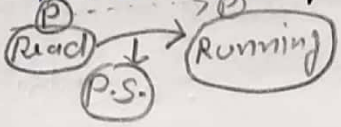
Q=②

- Process scheduling
- purpose of scheduling
- Types of scheduling

Process Scheduling :⇒

- The act of determining which process is in the ready state, and should be moved to the running state is known as process scheduling.

In other word :-



→ The process scheduling is the activity of the process manager that Handles the removal of the running process from CPU and

→ The selection of another process on the basis of a particular strategy.

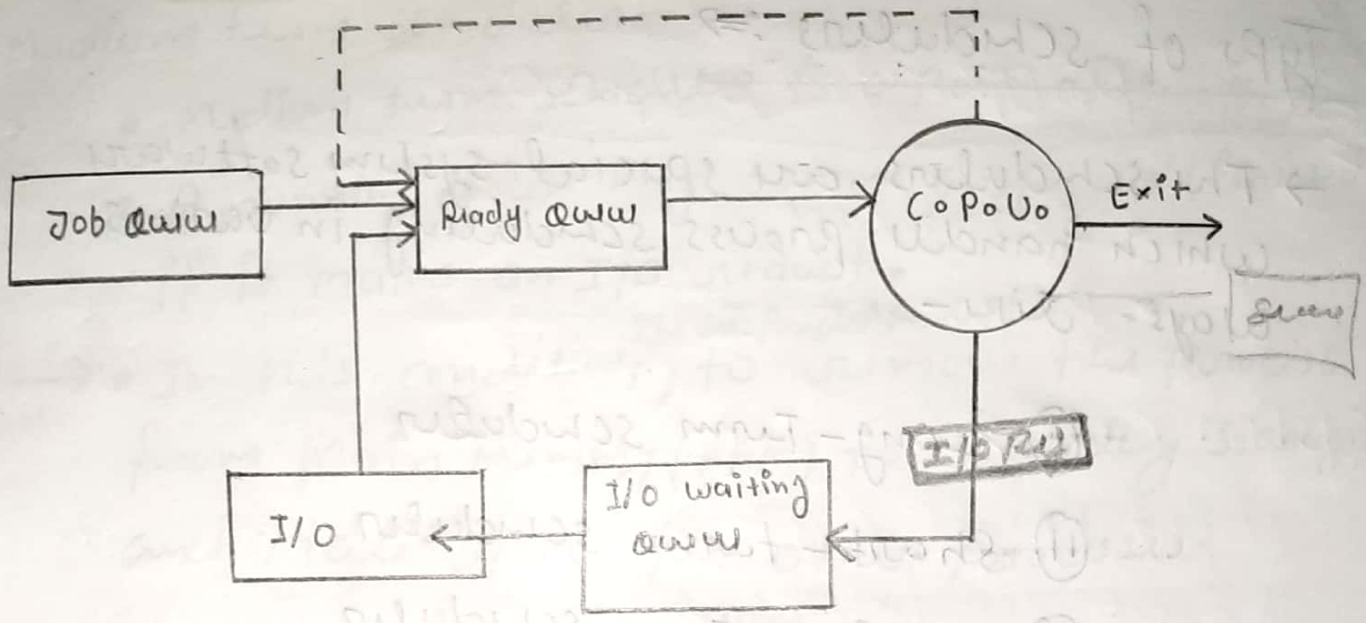
→ Process scheduling is an Essential part of a Multiprogramming operating system.

Process scheduling queues

- The Operating system maintains all PCBs in process scheduling queues.
- The OS maintains a separate queue for Each of the process state.

→ PCBs of all process in the same execution state are placed in the same queue.

→ When the state of a process is changed, so it's PCBs is moved from current queue to new state queue.



The operating system maintains the following Important Process scheduling queues:-

→ Job Queue:

This Queue keeps all the processes in the system.

→ Ready Queue:

Processes in the Ready state are placed in the Ready Queue.

→ Device Queue:

Processes waiting for a Device to become available are placed in Device Queue.

Type of schedulers :⇒

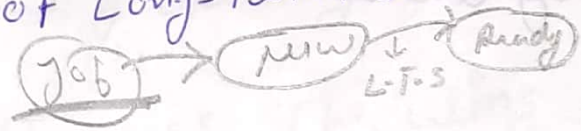
→ The schedulers are special system software which handle process scheduling in various ways. like-

(I) Long-Term scheduler
(II) Short-term scheduler
(III) mudium-Term scheduler

(I) Long-Term scheduler:

- It is also called a job scheduler.
- A Long-Term- scheduler determines which programs are Admitted to the system. for processing. and

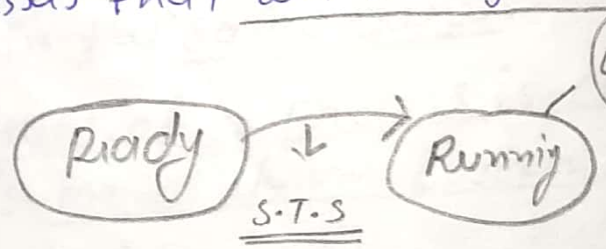
→ • when a process changes the state from new to ready, then there is use of Long-Term scheduler.



(II) Short-Term scheduler:

- It is also called CPU scheduler.
- It is the changes of ready state to Running state of the process.

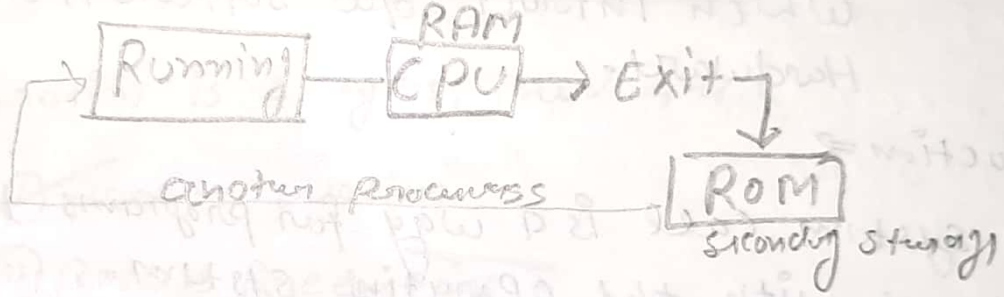
→ • CPU scheduler selects a process among the processes that are ready to execute.



(III) Medium term scheduler –

- Medium term scheduler is a part of swapping.
- A running process may become suspended (finished) if it makes an I/O request.

→ • In this condition, the suspended process to remove the process from main memory (RAM) to secondary storage and make space for other processes.



Q=3 Operation on process

- System call
- Process operation

- System call ⇒

Note ⇒ Kernel is the process system part which interact b/w softwares and Hardware.

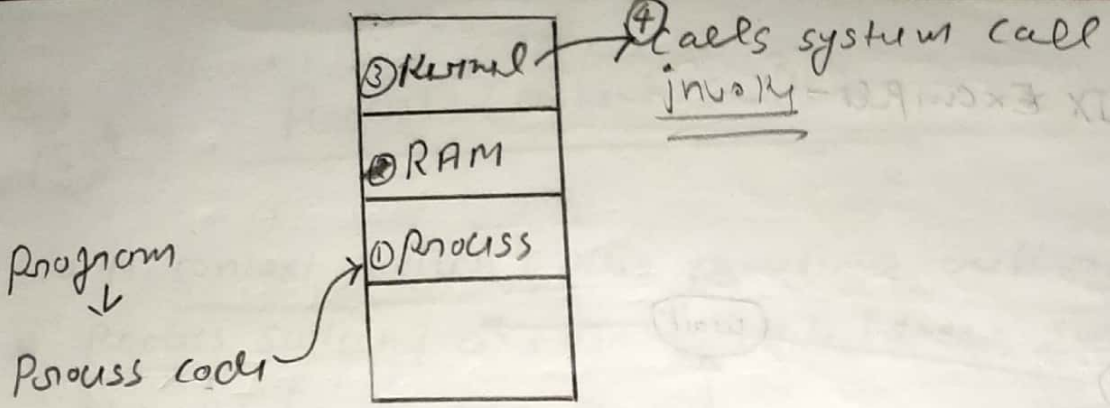
Introduction ⇒

- A system call is a way for programs to interact with the operating system.
- A computer program makes a system call when it makes a request to the operating system's kernel.
- The system call provides the services of the operating system to the user programs via (through) Application program Interface (API).

Types of system calls —

There are 5 different categories of system calls:-

1. Process control - End, abort, create, terminate, allocate, Load, Execute, Fork, wait etc.
2. File management - Read, write, open, close, create file
3. Device management - Read, write, reposition, lock etc.
4. Information - get pid, attributes, get system, and data etc.
5. Communication - pipe, create/delete connection etc.

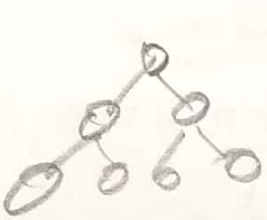


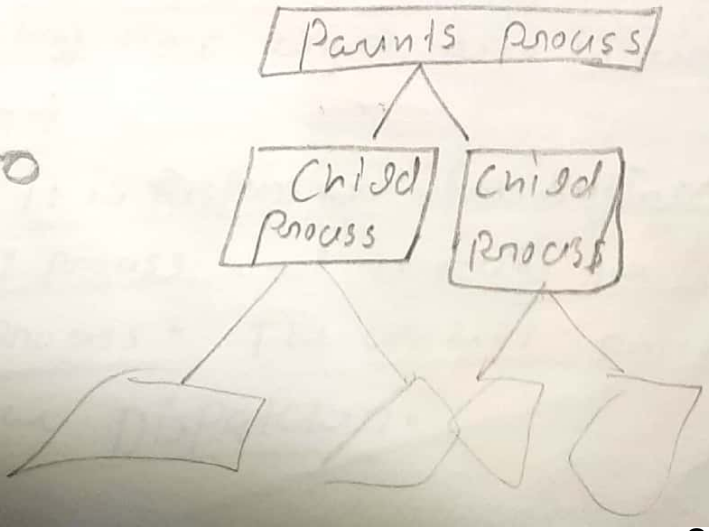
Operation on Process

- Process operation is 4 types, that are
  - (I) Process creation
  - (II) Process Preempting
  - (III) process blocking
  - (IV) Process Termination

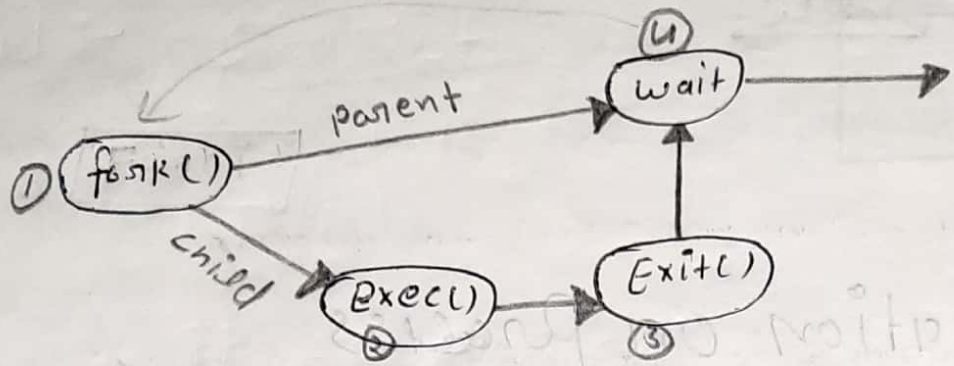
(I) Process creation :-

- A user requests, and already running process can create new program or process.
- Parents process creates Children process using a system call which is turn create other process forming a Tree of process.





Ex UNIX Example:-



- fork() system call creates new process
- Exec() system call used After a fork() to replace the process memory space with a new program.

(11) Process Termination:-

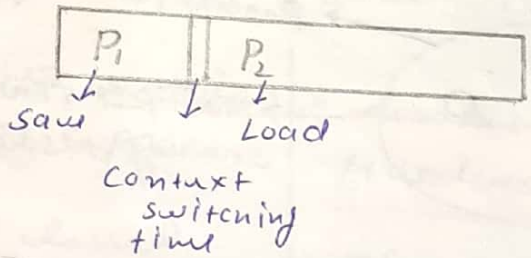
- Process Termination is an operation in which a process is terminated after the execution of it's last instruction (Exec()).
- The Process resources are deallocated by operating system.

Exit() → The Exit() system call is used to terminate the process after exec() system call.

Q=4

Process: Context switch

- A context switch (Also sometimes referred to as a process switch / a task switch) is the switching of the CPU from one process or to another.
- saving the context of 1 process and loading the context of another process is called context switching.



Some points ⇒

→ Each and Every time when process is moving from one state to another, the context of the process will change.

→ minimum process required for context switching.
Explain → Round Robin

→ The context switching will also take some time, so if the context of the process is more then context switching time will also increase which is undesirable.

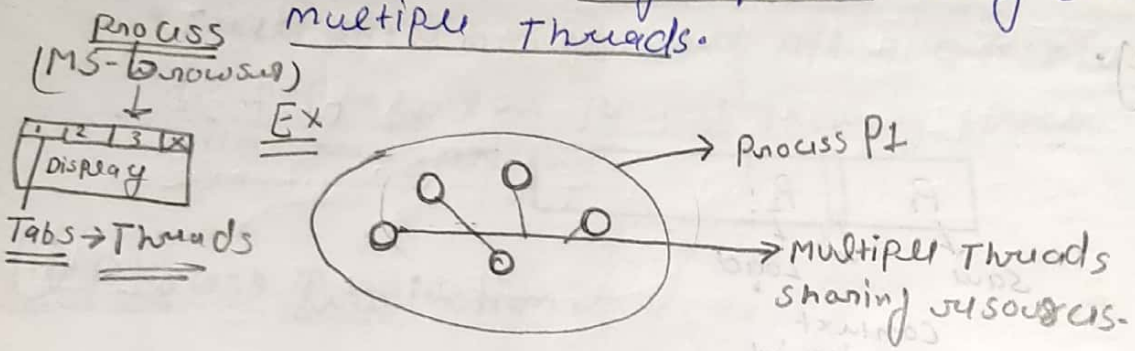
Dispatcher →

it is Responsible for saving the context of 1 process and loading the context of another process. The context switching is done by the Dispatcher.

Q=5 # Thread in OS (Lightweight Process)

Thread :- Smallest unit of processing that can be performed in an operating system.

It Always Exists within a process that means single process may contains multiple Threads.



Types of Threads:

Two types of thread as follows:-

(a) User Level Thread

(b) Kernel Level Thread

Advantage of Threads :-

- Responsiveness
- Faster context switch.
- Effective utilization of multiprocessor system.
- Resource Sharing.
- Communication b/w multiple Threads
- Enhanced Throughput of the system.

User Level Threads VS Kernel Level Threads

User Level Threads	Kernel Level Threads
• User Thread implemented by users.	• Kernel Thread implemented by O.S.
• O.S. does not recognized user level Threads.	• Kernel threads are recognized by O.S.
• Implementation of user Thread is Easy.	• Implementation of ~~user~~ kernel Level Thread is complicated.
• Context switches requires No Hardware support.	• Context switches requires Hardware SUPPORT is needed.
• If one user level Thread perform blocking operation then Entire Process will be blocked.	• If one kernel level Thread perform blocking operation then another Thread can continue execution.
• Ex: JAVA Thread	• Ex: Window Solaris.

Q=6 V.V. Most

CPU Scheduling

CPU Scheduling :-

- A Process Execution consist of a cycle of CPU Execution (Ready/New) and I/O Execution (Running).
- Normally Every process begins with CPU burst (E.T) that may be followed by I/O burst, then another CPU burst and then I/O burst and so on. Eventually in the last will End up on CPU burst.

Printer → अगर Printer (Process of CPU) CPU पर ज्यादा Work करता हो तो उसे CPU bound process कहते हैं और यदि Printer I/O burst पर ज्यादा Work करता हो तो उसे I/O bound Process कहते हैं।

CPU bound Process :-

There are those Processes which requires most of the CPU.

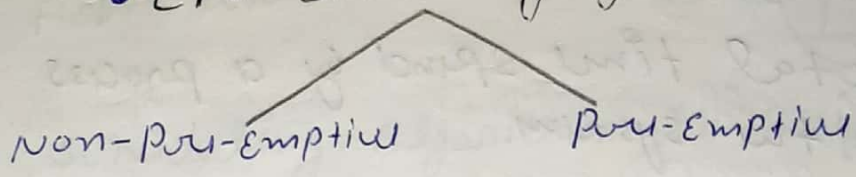
I/O bound Process :-

There are those processes which requires most of the time on I/O device or Peripherals devices.

Conclusion :-

A good CPU scheduling idea should Choose the mixture of both so that both I/O devices and CPU can be utilized Efficiently.

• CPU scheduling Types



Non-Pre-emptive :-

- The scheduling in which a running process can not be interrupted by any other process is known as Non-Preemptive scheduling.
- Any other process which enters the queue has to wait until the current process finishes it's CPU cycle.

Pre-emptive :-

- Preemptive scheduling means once a process started it's execution, the currently running process can be paused for a short period of time to handles some other process of higher priority.

• CPU Terminology

→ Burst/Execution/running time :-
is the time process requires for running on CPU.

→ waiting/Fairness time :- W.T. = T.A.T. — B.T.
time spend by a process in ready state waiting for CPU.

→ Arrival time :-
when A process Enter Ready state.

→ Exit time :-
when A process complete Execution and exit from CPU.

→ Turn-around time :-

Total time spend by a process in the system

$$T.A.T. = E.T. - A.T. = B.T. + W.T.$$

→ Response time :-

Time b/w a process enters Ready state and get scheduled on the CPU for the first time.

SCHEDULING

06.1 ALGORITHM

- The need for scheduling Algorithm arises ~~or achieve a~~ from the requirement for most modern systems to perform
  - multitasking -
    Execute more than one process at a time.
  - multiplexing -
    Transmit multiple flows simultaneously.

→ Scheduling Algorithm types :-

1. First come First Served (FCFS)
2. Round Robin (RR)
3. Shortest Job First (SJF)
4. Priority Scheduling

(1)

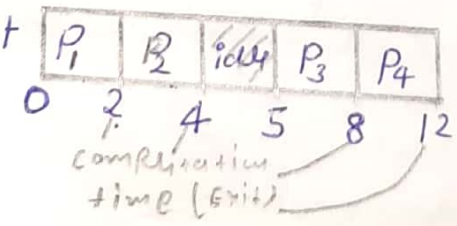
- First come First served :-
  - simplest scheduling Algorithm. It assign cpu to the process which Arrival first.
  - Easy to understand and can easily be implemented using queue data structure.
  - Always Non-Pre-emptive in Nature.
  - poor in performance as Average wait time is high.

(Criteria - Arrival time
Mode - Non-Preemptive)

Ex-

Process No.	Arrival Time	Burst Time	Complet. Time	TAT	WT	RT
P_1	0	2	2	2	0	0-0=0
P_2	1	2	4	3	1	2-1=1
P_3	5	3	8	3	0	5-5=0
P_4	6	4	12	6	2	8-6=2

Gantt Chart



- CT - AT = TAT
- TAT - BT = WT
- Avg. TAT = $\frac{2+3+3+6}{4}$
- Avg. WT = $\frac{0+1+0+2}{4}$

Convoy Effect :

Smaller process have to wait for long time for bigger process to Release from CPU.

(2) Shortest Job First (SJF) scheduling :-

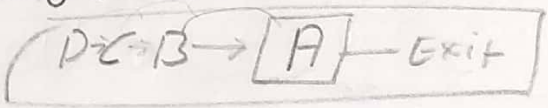
- Out of all available process, CPU is assigned to the process having smallest Burst Time requirement (No Priority, No Similarity).
- If time is a tie, FCFS is used to break tie.
- Can be used both with Non-Pre-emptive and Pre-emptive approach.
- Pre-emptive version (SRTF) is also known as optimal as it generate minimal average waiting time.

Q-7

Process Synchronization

Introduction →

- Process Synchronization occurs when several processes RUN in an operating system.
- There are formation of queues which form when multiple processes are to be Executed.
- During execution of the processes invading a serialized part of the program.
- The process synchromization is also known as serialization.



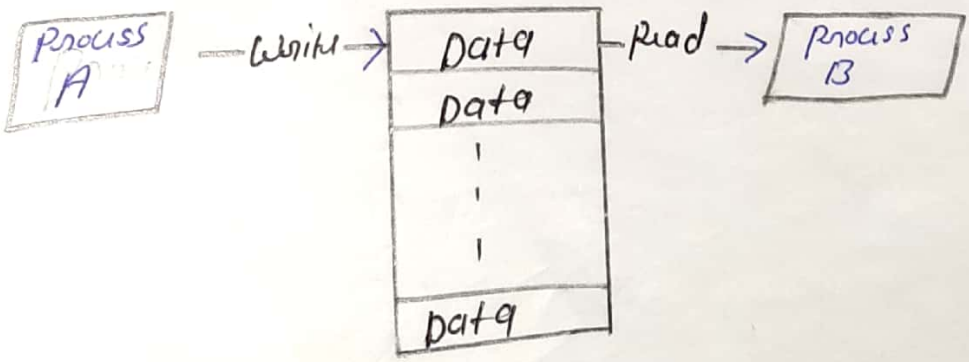
Note:-

Some of processes share resources due to which problems like data inconsistancy may arises-

For Example:-

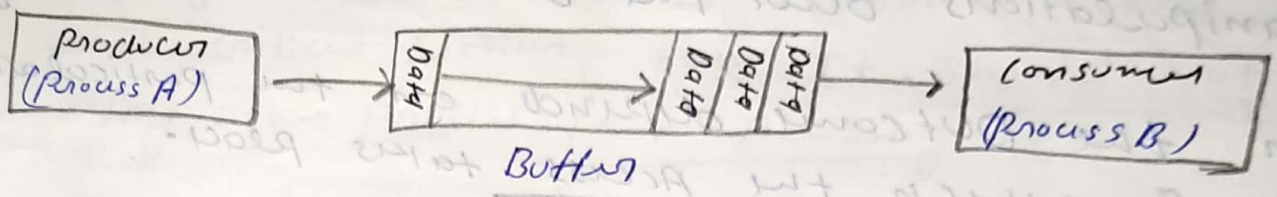
- One process changing the data in a memory location where Another process is trying to Read the data from the same memory location.

→ It is possible that the data ready by the second process will be Erroneous.



Producer & Consumer Problem :-

- In process synchronization there is a thumb Rule which states that • production of Information is done by the Producer and the Consumption is done by the Consumer process.



- Problem → To Ensure that the producer should not add Data when the Buffer is full.
  → And the Consumer should not take data when the Buffer is Empty.

Type of process Synchronization :→

Process synchronization is categorized in two following types:

(I) Independent Process :- Execution of one Process does not Effects the Execution of other process.

(II) Cooprative Process → Execution of one process Effects the Execution of other process.

Critical-section problem

Race condition :-

~~Several processes~~

- Several processes access and process the manipulations over the same data concurrently, then the outcome depends on the particular order in which the access takes place.

Critical-section problem :-

- While process synchronization some problems may Arise due to conflict between different processes.
- While multiple processes are sharing the segment of code, A conflict may arise →

  If one process updates the code while other processes are Reading it. This conflict as a Critical section problem.

Solution of Critical section problem ⇒

Any solution of critical section problem the three following Nuds are mandatory.

(I) Mutual Exclusion
(II) Progress
(III) Bounded waiting.

(I) Mutual Exclusion :-

When a process is in the critical section Execution, NO other process can be executed at the same time.

(II) Progress :-

- Progress means that If a process is not using the critical section, then It should not stop Any other process from Accessing it.

→ In other word, any process can Enter A critical section If it is free.

(III) Bounded waiting :-

- Bounded waiting means that Each Process must have a limited waiting time.

It should not wait Endlessly to Access the critical section.

Concept of Semaphores

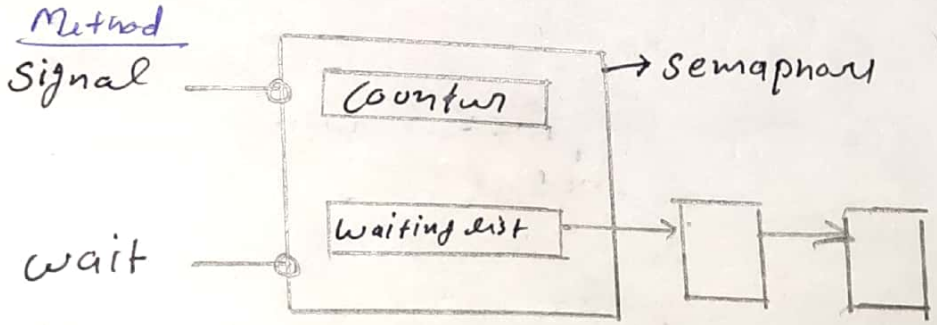
- Semaphores is A very important and popular tool used for process synchronization.
- Semaphore is used to protect any resources such as global shared memory that needs to be accessed and updated by many processes simultaneously.

Semaphore Acts as a guard/Lock on the Resource ⇒

- When ever a process needs to Access the resource. It first needs to take permission from the Semaphore.
- Semaphore give permission to access a resource if Resource is free otherwise process has to wait.

The Semaphore is implemented by variables like ⇒

Counter, a waiting list of processes and two methods (eg. functions) signal & wait. with integer values.



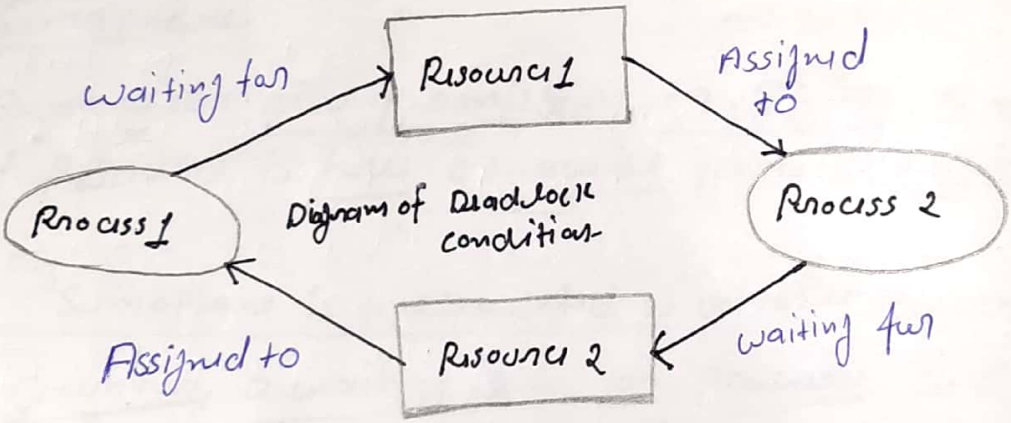
- The Semaphore is accessed by only two indivisible Operations known as wait and signal.

UNIT – III

Introduction :-

- Deadlock is a situation where a set of processes or two or more processes are each waiting for each other to release a resource.

- Deadlock occurs when a set of processes are blocked because each blocked process holding a resource and waiting for another resource and this resource acquired by other process.



System model :→

→ (i) Every process will request for the resource.

→ (ii) If granted then, the process will use the resource.

→ (iii) Process must release the resource after use.

Necessary conditions of Deadlock :-

Deadlock can arise if following four conditions hold simultaneously:

(a) MUTUAL EXCLUSION :- Lux

eg YT-video One or more than one resource are non-sharable and only one process can use at a time.

(b) Hold and wait :-

A process is holding at least one resource and waiting for resources.

(c) No Preemption :-

A resource can not be taken from a process unless the process releases the resource.

(d) Circular wait :-

A set of processes are waiting for each other in circular form.

METHODS FOR HANDLING DEADLOCK :⇒

(I) Prevention → [4 condition of deadlock]

- Prevention means design such a system in which violate at least one of four Necessary conditions of Deadlock and Ensure independence from deadlock.

(II) Avoidance → [safe data]

- A system maintains a set of data using which the system takes a decision whether to Entertain [idle, Think] a new Request or Not, to be in safe data.

(III) Detection and recovery →

- Here we wait until Deadlock occurs and Then do prevention to handle it once occurred.

(IV) Ignore the problem →

- we ignore the problem as if does not exit.

UNIT- IV

Q.1 What is Device management?

Introduction :-

- Device management controls Peripheral devices by sending them commands in their own proprietary language. [own set of language]
- The software Routine that knows how to deal with each device is called a Driver.
- And Operating system organizes drivers for the Peripherals Attached to the Computer.
- When A New Peripheral is Added, that device's driver is installed into the Operating System.

Motive of Device management :-

- The main motive of Device management is to Ease the usage of hardware.
- A Device manager Controls the processing of drivers.
- The Drivers function as a pathway between the hardware and software.

Note :→ Operating system maintains the system for easy and secure usage. and It defines data as files. And A Driver translates the files into streams of bits to be kept on the storage devices at specific locations.

Q-2 The techniques for device management in operating system →

1. Dedicated Devices
2. shared Devices
3. Virtual Devices
4. Storage Devices
5. Input an output devices
6. Buffering process

1. Dedicated Devices

- There are different kinds of devices Among which are the dedicated devices. From the name only we get that dedicated devices are Allotted for dedicated work.
- The devices that are assigned for only one job at a Entire time are called dedicated devices.
- These devices run for the Entire time serving only a single job when it is Active.

Example- Tap Drives, Plotters and printers.

→ These kind of dedicated devices are easy to use but they take a lot of time to Execute a list of commands Since They work Each Commands at a time.

② Shared Devices

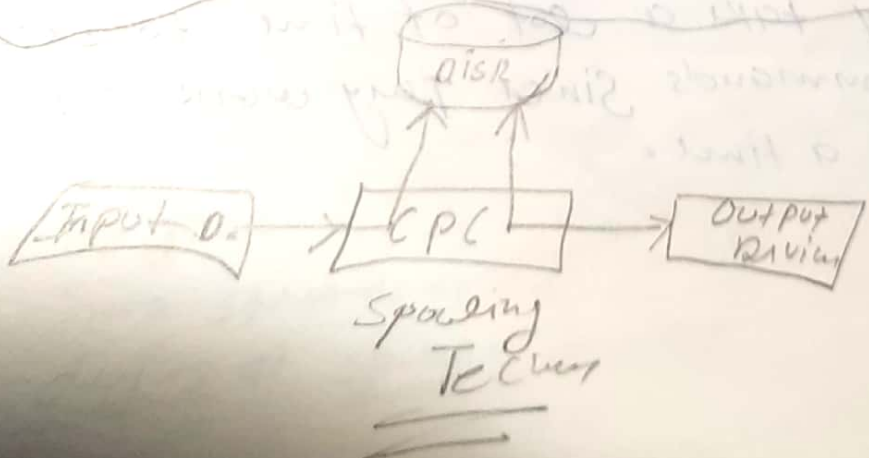
Introduction :-

- Shared devices are the devices that can be shared among several processes. They can be assigned to more than one process at a time.
- These devices include direct Access storage devices such as magnetic disks and optical disks.

How do the shared devices work :-

- With the Advancement of technology computer science is trying to Adopt new ideas day by day and trying to make the computing process easier for people.
- Sharing is an important part of computer network. Not only Exchange of files or data but also the devices can be shared Among many computers through Network and Internet support as well.
- Shared devices can be used and Accessed by many people at the same time. with the help of LAN (Local Area Network). in Any workplace.

Example - Shared printer, shared scanner, shared file or document and so on.

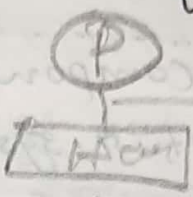


(3) Virtual Devices

Introduction :-

- In general virtual devices are dedicated devices which are transformed into shared devices through the technique of **spooling**.

Spooling Technique E.g. →



- Printers and faxes use spooling technique by routing all the printer requests to a disk. (Send something)
- And printer writes output only when the output is completed and the printer is ready to printout the desired document.

Note →

- A virtual device is the combination of the other kind of devices namely dedicated devices and shared devices.
- There is no general method to install the virtual devices. Also A device ID is not Exposed by the virtual devices.

Advantages →

1. There will be no need to have a second system while testing the OS.
2. More Enhanced User Experience.
3. Very good quality of service.

(4) Input or Output Devices

- A computer system consist of various components and It is worth mentioning that most of these components are said to be peripherals.

- These components are used to store data into the computer system and also get the processed data in the end.

→ and These devices are known as Input or output devices and they are available in various type.

Input devices –

- The input devices are chiefly used to feed data and Information into the computer.
- These devices are Attached to the system and are used to provide Essential Information and Instruction to the system with which the computer Actually works.

And this works as the Raw Materials for it to process.

Output Devices –

- The devices that are used to get the processed information are known as output devices.
- These devices are also Attached with the system Externally.

(5) Storage devices

- Storage devices are very important components of any computer system. And these days most of the Advanced computer systems are making use of a variety of storage devices to Enhance their functions

- The storage devices are nothing but very special devices that are used to store the data can be very safe and secure for longer period of time.

Most common storage devices :-

- The most common devices for storage of data and Information can include

  - Floppy Disks
  - Compact Disks Recordable (CD+R)
  - Compact Disks Rewritable (CD+RW)
  - Digital Versatile Disks (DVD)
  - (DVD+RW) and (DVD+R)
  - ZIP Disks.

⑥ Buffering

- The Buffering refers to the process of Pre-loading data into a reserved area of memory.
(Soft Area)
- Buffering is the process of downloading a specific Amount of data.

→ This process of operating system starts before starting to play the music or the video that downloads.

⇒ Even if someone is an about to download a movie, the process of buffering starts before the person starts to watch the movie.

More About Buffering :-

- Buffering is a process of reading or writing data from HDD.
- It involves a long time for doing this process of reading and writing.
- It is Improves the speed for data processing.

Q (3) Writes the short notes on the following:

1. Disk structure
2. Disk scheduling
3. Disk managment

(1) Disk structure

- Most of the Computer users these days are becoming more and more careful about selecting the best brand of Computer systems.
- The memory structure is one of the most important aspects of these computers systems.
- Various types of Disks are being used by the users to strengthen their system and

Also to make and store that data and Information for a longer period of time.

Disk sector and Disk Tracks :-

Disk Tracks - Disk structures are round in shape so the Disk Tracks are also strong circular paths which Run in the circular shape inside the Disks. (Round Robin / Circular paths)

Disk ~~Fac~~ sector - The Disk sectors are smaller Areas inside the Disks where the data is Actually stored inside the disk.

- The Disk sectors are formatted to contain a good Amount of data which is generally 512 bytes of data

(2) Disk Scheduling

Introduction :-

- Though Operating system has many important function one of the most important function which it serves is using the hardware efficiently.
- For the disk drives, completing this task Ensures Fast Access Time Along with Disk Bandwidth.
- Access Time - is composed of two Elements as
  (I) Seek time
  (II) Rotational Latency.
  (III) Transfer time

(I) Seek Time -

- Seek Time is the time taken for a disk drive to locate the Disk Arm (Arm on the Disk) to a specified Track where the data to be read or write.
- So the disk scheduling Algorithm that gives minimum Average Seek Time is better.

(II) Rotational Latency -

- Rotational Latency is the time taken by the desired sector to Rotate into position.
- So the Disk Scheduling Algorithm that gives minimum Rotational Latency is better.

(iii) Transfer time :—

- When the data is transferred in bytes and further ~~is dt~~ Transferred Data is divided ~~into~~ by Length of time between the First steps of transfer and the last steps of transfer.

- Access Time and Bandwidth can be Easily Rectified with the help of scheduling.

(iv) Access Time →

Access Time is →

Disk Access Time = Seek Time + Rotational Latency + Transfer Time.

Disk scheduling Algorithm :—

- The process of scheduling mainly Focuses on the Servicing of the Disk Input and Output requests in proper order.
- scheduling Algorithms are as

(i) First Come First Served (FCFS)

(ii) Shortest Seek Time First (SSTF)

(iii) scan

(iv) Circular scan (Track)

UNIT - V

Q.1 What is Information management and File system with it's type?

Information management –

- It means the OS manages the information and data.
- In other word, managing the information from which or to which it should be sent/Received.

FILE SYSTEM –

- File system is a method for storing and organizing computer files and data.
- File system in OS helps the track of Information, It's Location and Everything.
- It may use a data storage device such as Hard DISK, or CD-ROM and Involve maintaining the physical Location of the files.
- They typically have Directories which Associates filename with files.

Example – FAT – File Allocation Table

NTFS – New Technology File System

Etc. –

Types of file systems—

(1) Disk File systems—

- It is file systems designed for the storage of files on a data storage and A disk drive which might be directly or Indirectly connected to the computer.

Examples— FAT, FAT32, NTFS, HFS, etc.

(2) Flash File systems—

- It is a file system designed for storing files on Flash memory devices.

Examples— JFFS 2.

(3) Data-Base File systems— [Hierarchical Structured memory]

- It is a new concept of file management in which Instead of Hierarchical Structured management.

(4) Transactional File system—

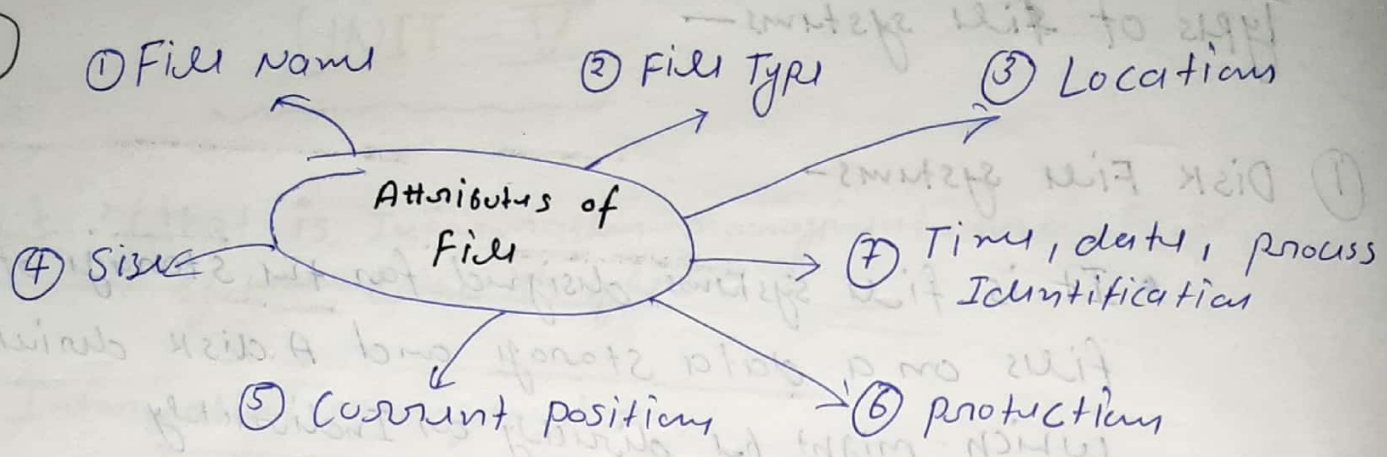
- This is a special kind of File system in which it Logs Events as transaction to files.

(5) Network file-System—

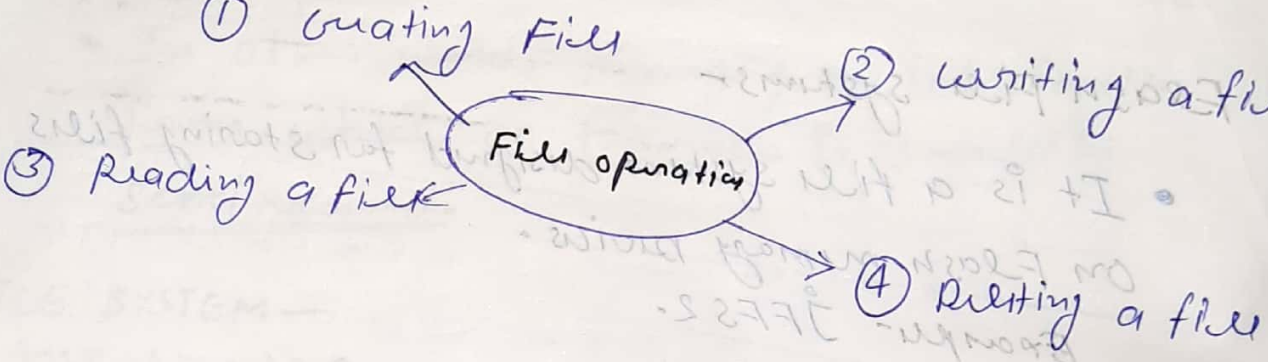
- It Acts as a client for a Remote File Access protocal, providing Access to files on a server.

Example— NFS, FTP, etc.

(1)



(2)



File Access METHODS

- Information in files could be accessed in many ways as:

  (a) Sequential Access
  (b) Direct Access
  (c) Indexed sequential Access

(a) Sequential Access-

In this, the information of a file is processed in order. And as the operations on a file is read and writes one by next portion of the file.

(b) Direct Access :-

- In this, There is no order of storage of files.
- It Allows Arbitrary blocks to be read or write. And this method is very useful to Access the Large Amount of Data in Immediate way.

(c) Indexed Sequential Access :-

We First find or search the Index and then by using this Index Reaches to the desired Records.

Q-2 Directory structure

File systems are very large. And Files have to be organized. In two way as-

① Partition

1 → • The file system is divided into partitions. Default there is at least one partitions.

• Partitions are nothing but virtual disks with Each partition considered as a separate storage device.

② Directory

2 → • Each partition has information about the files in it. This information is nothing but A Table of content. It is known as Directory.

➡ The directory maintains information about the Name, Location, size, and type of All files in the partition.

• There are some operation can be performed on Directory such as:—

- Create a File
- Delete a file
- List a directory
- Rename a file
- Traverse the File system

Types of Directory :-

(I) Single level D.
(II) Two-level D.
(III) Tree-structured D.