# Assignment : 2

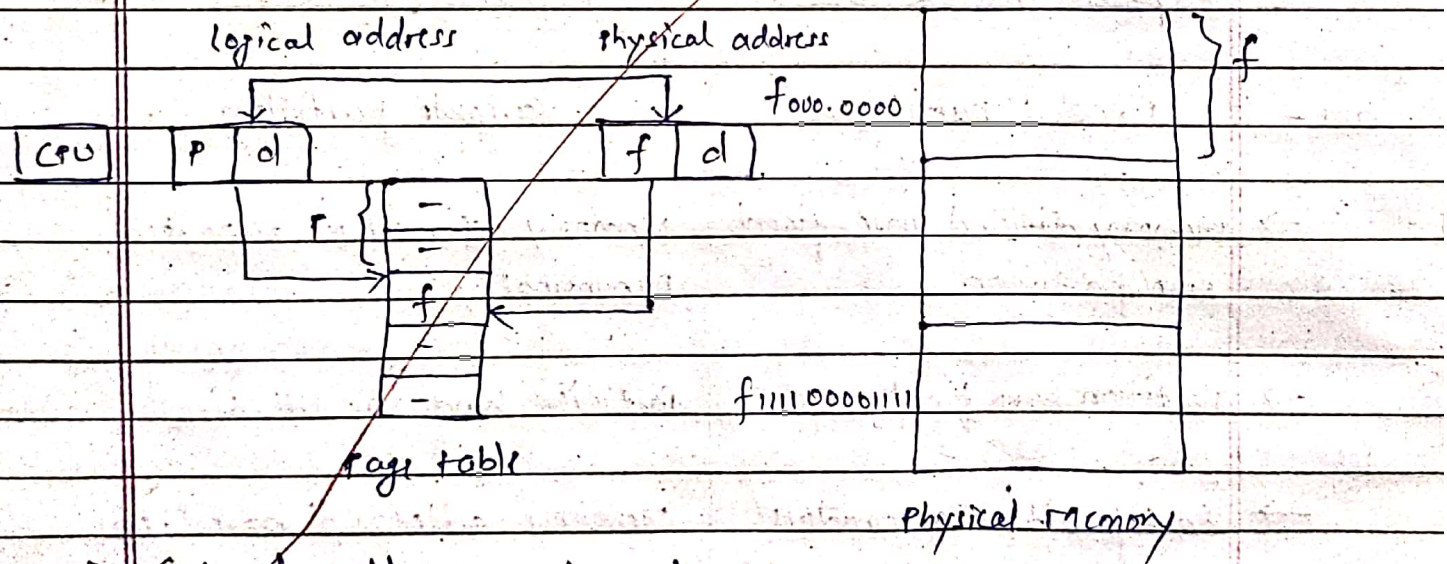
Date
Page

1 What is paging? Explain the conversion of virtual address in paging.

Ans. Paging : Paging helps manage memory in computer by moving process from secondary storage to main memory in pages.

⇒ Each process get divided into smaller part known as page main memory gets divided into fixed sied section called frames. Pages of a process are stored in processing within the memory.

Conversion of virtual address in paging.




→ Virtual address components -
- split into two parts 1) Page Number 2) Offset
- Page no identifies the page in virtual memory
- offset point is the specific byte within the page.

→ Page table -
- Used by MMU for address address translation.
- Contain entries for each page in virtual memory.
- Each entries includes a frame no. indicate the physical memory.

* Conversion steps —
- MMU uses page no. (p) to find the page table entry.
- Retrieve the page no. from this entry
- Converts it with d to form physical address.

⇒ Example -
→ Consider a virtual address page 10 offset 12
→ If the page table shows that page no. coresides in frame 5.
→ The physical address would then formed as 5012

Qno 2. Compare multiprogramming with fixed partitioning and multiprogramming with variable partition with diagram. Explain internal and external fragmentation with example.

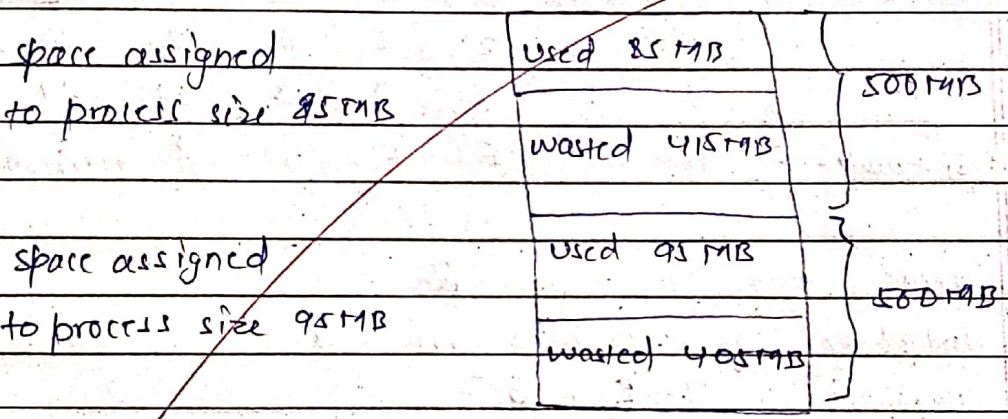
ans -

| fixed partition | Variable partition |
|---|---|
| → Memory divided into fixed size partition | Memory shared or allocated dynamically. |
| → Partition have fixed size. | Partition have variable size |
| → Each partition accomodate exactly one process | Memory allocated based on process usage |
| → Partition boundaries are not movable | They can change dynamically |
| → May lead to ineffective utilization due to fixed size | Better utilization as memory is allocated used on needs. |

- Internal fragmentation : It happen when a process is allocated more memory than it actually needs, leading caused space within the allocated memory block.

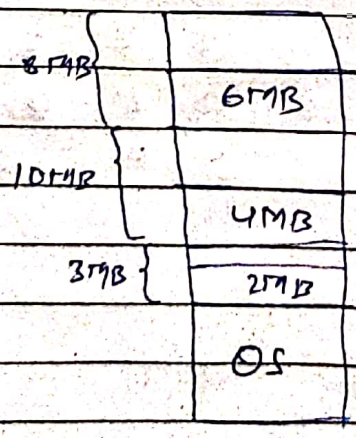
Eg- Two process are listed one with a size of 85 mb and other size with 95 mb.

⇒ Each process is allocated a space of 500MB This extra space is wasted memory because its more than what each process need.




- External fragmentation : It occure when there is enough total memory space available to satisfy a process memory req. but this space is ~~also~~ scattered non-contigeous moving it usable for further allocation.

Eg-



Date
Page

Qno.3. Consider the reference string 1, 2, 1, 0, 3, 0, 4, 2, 4 and no. of frames to be 3. Apply LRU and FIFO page replacement algorithm and compute hit ratio and miss ratio.

ans – References string – 1, 2, 1, 0, 3, 0, 4, 2, 4
Frame = 3 (3 pages can be in memory at a time per process)

**FIFO**

| | 1 | 2 | 1 | 0 | 3 | 0 | 4 | 2 | 4 |
|---|---|---|---|---|---|---|---|---|---|
| | | | | 0 | 0 | 0 | 0 | 2 | 2 |
| | | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 |
| | 1 | 1 | 1 | 1 | 3 | 3 | 4 | 4 | 4 |
| Page Fault | F | F | | F | F | | F | F | |

Here, Page fault are 6
Total no. of references are = 9
Page Hit = 9 – 6 = 3
Hit ratio = $\frac{3}{9} = \frac{1}{3}$

Miss ratio = $1 - \frac{3}{9} = \frac{6}{9}$

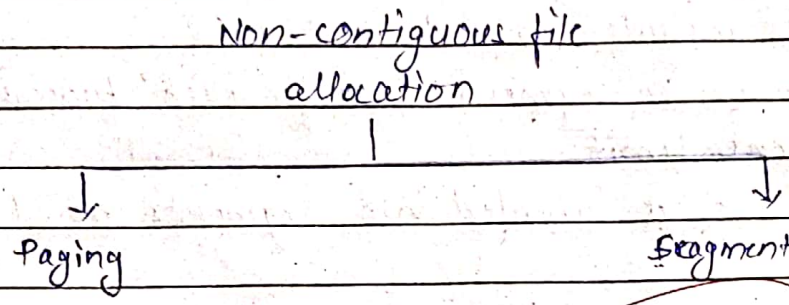
**LRU**

| | 1 | 2 | 1 | 0 | 3 | 0 | 4 | 2 | 4 |
|---|---|---|---|---|---|---|---|---|---|
| | 1 | 1 | 1 | 1 | 3 | 3 | 4 | 4 | 4 |
| | | 2 | 2 | 2 | 2 | 2 | 3 | 2 | 3 |
| | | | | 0 | 0 | 0 | 0 | 0 | 2 |
| | F | F | | F | F | | F | F | |

Here, Page Fault are 6
Total no. of references are = 9
Page Hit = 9 – 6 = 3
Hit ratio = $3/9 = \frac{1}{3}$
Miss ratio = $1 - \frac{3}{9} = \frac{6}{9} = \frac{2}{3}$

Q.no 4. Explain non contiguous file allocation technique along with advantage and disadvantage.

ans-




- Paging : Physical memory divided into fixed size frame while logical space divided into pages.

→ Each page and frame are of same size
→ OS maintain a page table for each process
→ One free frame list is maintained
→ When a process receives its pages are moved from frame in memory

- Advantages :-

→ Simplified memory management.
→ Reduces external fragmentation.
→ Efficient memory utilization.
→ Allow for efficient use of memory resource

- Disadvantages

→ Higher effective access time with additional memory allowance.
→ May lead to increased memory wastage.
→ Limited flexibility in memory allowance.

- Segmentation : Process divided into segments of variable sizes representing a logical address space.

→ Segments include main data structure
→ OS maintain a segment table for each process base address segment limit
→ logical address divided into segments No. (offsets)

- Advantages :

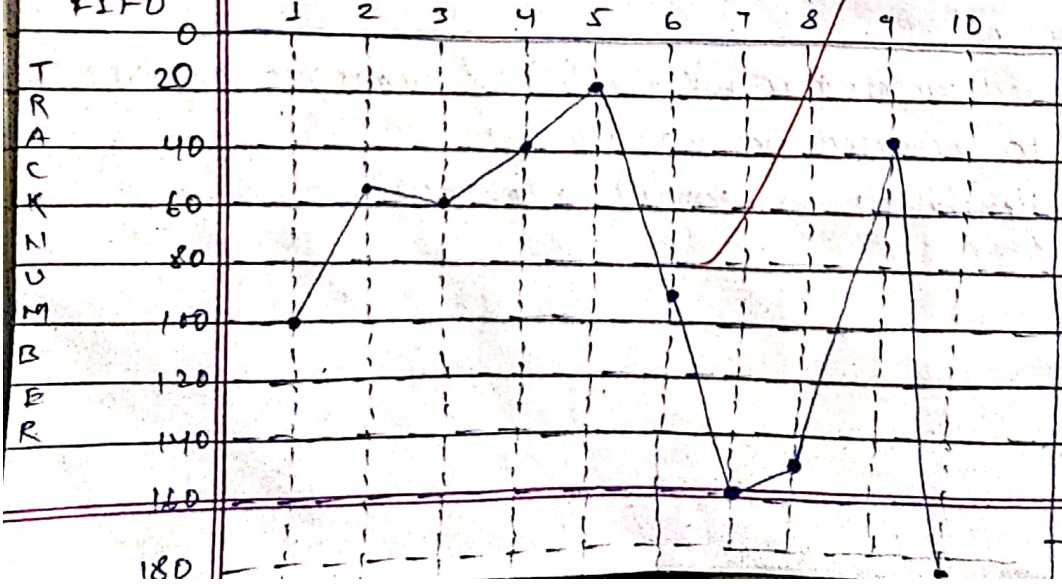
→ No internal fragmentation
→ Flexible memory management
→ support sharing and protection

- Disadvantages :

→ External fragmentation issue
→ Complex management
→ Overhead in table management

Q no 5. Consider a disk with 200 tracks and the queue has random request from different process in the order : 55, 58, 39, 18, 90, 160, 150, 38, 184. Initially arm is at 100, Find the average seek length using FIFO and SSTF algorithms

FIFO




| accessed Next track | transact No. of track |
|---|---|
| 55 | 45 |
| 58 | 3 |
| 39 | 19 |
| 18 | 21 |
| 90 | 72 |
| 160 | 70 |
| 150 | 10 |
| 38 | 112 |
| 184 | 146 |
| Avg | 55.3 |

- Total head movment :

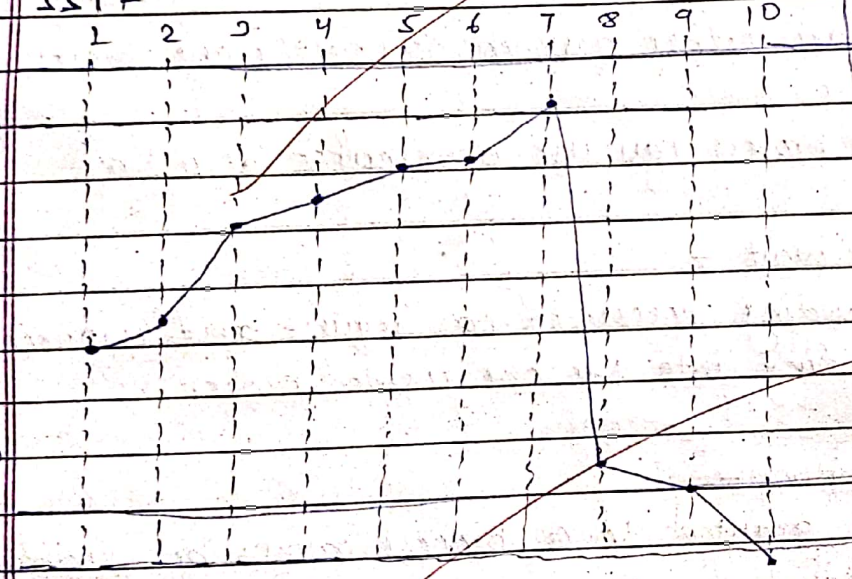
= (100 - 55) + (58 - 55) + (58 - 39) + (39 - 18) + (90 - 18) + (160 - 90) + (160 - 150) + (150 - 38) + (184 - 38)

= 45 + 3 + 19 + 21 + 72 + 70 + 10 + 112 + 146

= 498 cylinders

∴ Avg seek length = $\frac{498}{9}$ = 55.34 ms

- SSTF




| Next track accessed | No. of track traversed |
|---|---|
| 90 | 10 |
| 58 | 32 |
| 58 | 3 |
| 39 | 16 |
| 38 | 1 |
| 18 | 20 |
| 150 | 132 |
| 160 | 10 |
| 184 | 24 |
| Avg seek length = | 27.5 |

- Total head movment :

= (100 - 90) + (90 - 58) + (58 - 55) + (55 - 39) + (39 - 38) + (38 - 18) + (150 - 18) + (160 - 150) + (184 - 160)

= 10 + 32 + 3 + 16 + 1 + 20 + 132 + 10 + 24

= 248 cylinders

∴ Avg seek length = $\frac{248}{9}$ = 27.56 ms

Date
Page

Qno 6. Define deadlock. what are the necessary condition for deadlock.

Ans - Deadlock occurs when a set of process are unable to proceed because each process is waiting for an event that only another process in the set can cause.

* Necessary condition for deadlock.

1) Mutual Exclusion -
- Resource are either assigned to exactly one process or available.
- Only one process can use a resource at a time.

2) Hold and wait -
- Process holding resources can request and resource while still holding into the one granted earlier.

3) No preemption -
- Resourcey granted to a process cannot be forcibly taken equally.

4. Circular wait -
- There must exist a circular chain of two or more process each waiting for a resource held by the next member of the van.

Qno 7. Explain the idea of deadlock prevention and avoidance in operating system.

ans - Dead lock prevention -

- Attacking mutual exclusion :-

→ Assign each resources to more than one process but not flexible for certain resources like pointer

- Hold and wait :-

→ Require process to request all resources they needed before starting execution.

→ Process allowed to access only if all needed resources are available but may not know all resources in advanced

- No preemption :-

→ Probably take away resources from one process allocates to another upon request but not flexible for certain resources like pointer.

- Circular wait :-

→ All resources and allow process to resources only in numerical order.

→ Process holding a resources cannot wait for a resources with a higher no. prevent circular wait

→ Ensures resources graph never from a cycle preventing deadlock.

Qno 8. Explain banker algorithm in details

ans - Banker algorithm is used to avoid deadlock and allocate resource safely to each process in the computer system The s-state examine all possible tests or activities before deciding whether the allocation should be allowed to each process it also helps the operating system to successfully share the resources between all the process

⇒ when working with a Banker's algorithm it requests to know about three things.

1) How much each process can request for each resource in the system. It is denoted by the [max] request.

2) How much each process is currently holding each resources in a system. It is denoted by the [Allocated] resource.

3) It represents the number of each resources currently available in the system. It is denoted by the [Available] resources.

* Advantage:

1) It contains various resources they meet the requirement of each process.

2) Each process should provide information to the OS for upcoming resource requests the no. of resource and how long the resource will be help.

3) The algorithm has a max resource attribute that represents each process can hold the maximum of resource in a system.

* Disadvantage:

1) It require a fixed no. of resources process and in addition process can be stored in the system while executing the process.

2) The algorithm does not longer allows the process to exchange its maximum needs while processing its tasks.

3) Each process has to know and start their maximum resource requirement in advance for the system.

4) The no. of resources requests can be granted in a finite time but the time limit for allocating the resource in one year.

Qno 9. Use banker's algorithm to answer the following question.

| Process | Max | Allocation | Available |
|---|---|---|---|
| | A B C D | A B C D | A B C D |
| $P_0$ | 6 0 1 2 | 4 0 0 1 | 3 2 1 1 |
| $P_1$ | 2 7 5 0 | 1 1 0 0 | 9 2 2 3 |
| $P_2$ | 2 3 5 6 | 1 2 5 4 | 11 5 7 9 |
| $P_3$ | 1 6 5 3 | 0 6 3 3 | 12 11 12 12 |
| $P_4$ | 1 6 5 6 | 0 2 1 2 | 13 17 17 18 |

(i) How many resources of type A, B, C, D, are there?

A = 4 + 1 + 1 + 3 = 9

B = 1 + 2 + 6 + 2 + 2 = 13

C = 5 + 3 + 1 + 1 = 10

D = 1 + 4 + 3 + 2 + 1 = 11

(ii) What are the contents of need matrix?

Need matrix:

| | | | | | | |
|---|---|---|---|---|---|---|
| $P_0$ | 6 0 1 2 | | 4 0 0 1 | | 2 0 1 1 |
| $P_1$ | 2 1 5 0 | | 1 1 0 0 | | 1 6 5 0 |
| $P_2$ | 2 3 5 6 | − | 1 2 5 4 | = | 1 1 0 2 |
| $P_3$ | 1 6 5 3 | | 0 6 3 3 | | 1 0 2 0 |
| $P_4$ | 1 6 5 3 | | 0 2 1 2 | | 1 4 4 4 |

(iii) Find if the system is in safe state ? if it is find the safe sequence.

ans- System is in safe state.
Safe sequence :- $P_0, P_2, P_3, P_4, P_1$

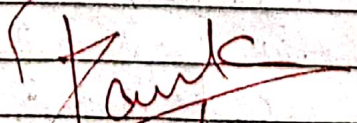
Q no 10. Define locality of reference, page fault, Demand paging and Parity bit.

ans- Locality of reference :-
locality of reference refers to the tendency of the computer program to access the same set of memory location for a particular time period.

On an abstract level there are two types of localities.
1) Temporary
2) Spatial.

* Page fault :-
→ Page fault ~~domain~~ dominate more like an error. A page fault will happen if a program tries to access a pieces of memory that does not exist in physical memory.

* Demand paging :-
It is a memory management approach where page are loaded into memory only when needed busy program execution.




* Dirty bit :-
A dirty bit is also known as ~~d~~ modified bit or write bit as a page that is used in computer system to indicate where a particular memory address.