D. Manogna
21B81A1294

1) What is page fault? Explain the steps involved in handling a page-fault with a neat sketch.

A) Page Fault:- A page fault will happen if a program tries to access a piece of memory that does n't exists in physical memory. The fault specifies the OS to trace all data into virtual memory management and then relocate it from secondary memory to its primary memory, such a hard disk.

Handling a Page Fault:-

The procedure for handling a page fault in OS:-

⇒ Firstly, an internal table for this process to assess whether the reference was valid or invalid memory access.

⇒ If the reference becomes invalid, the system process would be terminated. Otherwise, the page will be paged in.

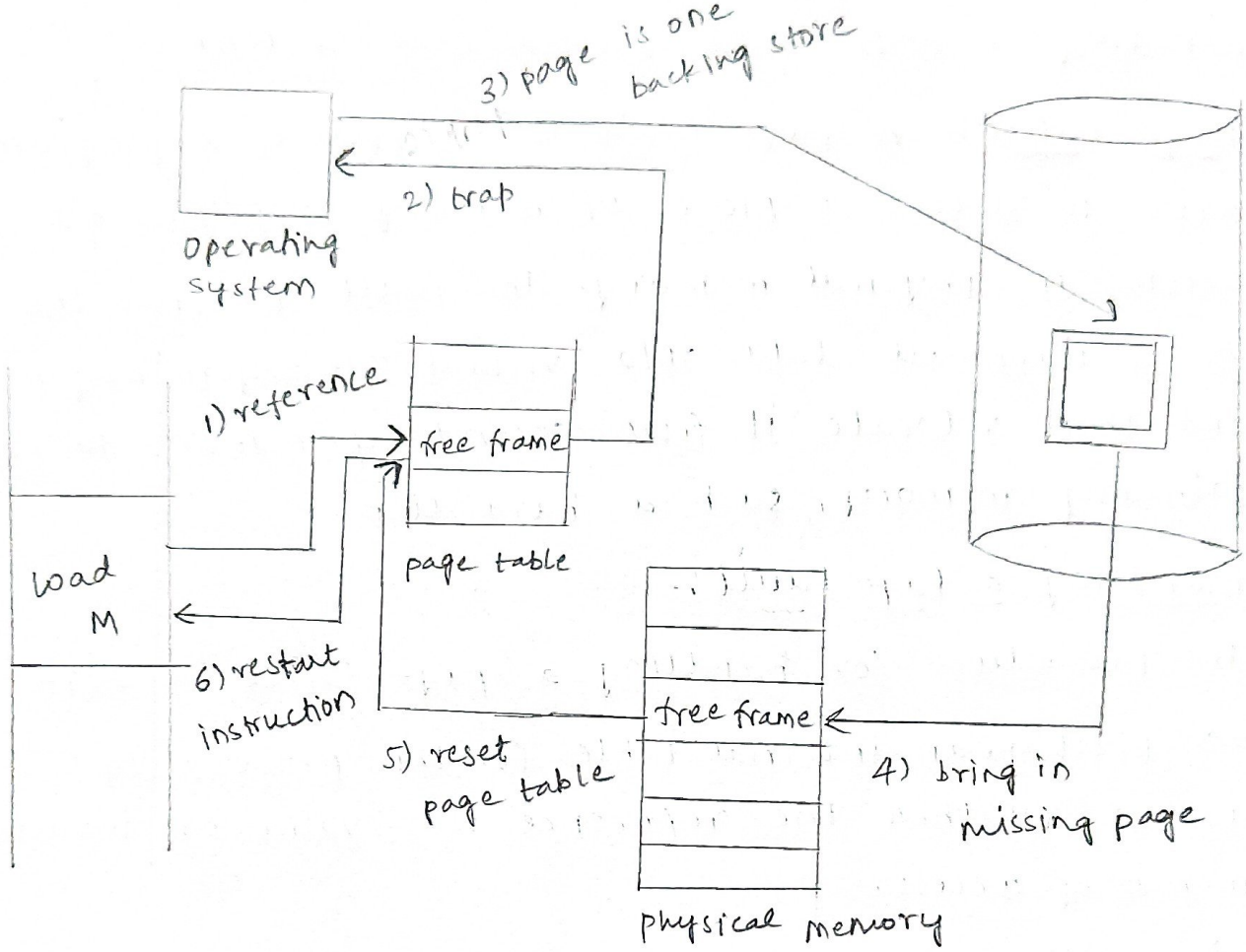
⇒ After that, the free-frame list finds the free frame in the system.

⇒ Now, the disk operation would be scheduled to get the required page from the disk.

⇒ When the I/O operation is completed, the process's page table will be updated with a new frame number, and the invalid bit will be changed. Now, it is a valid page reference.

⇒ If any page faults are found, restart these

steps from starting.

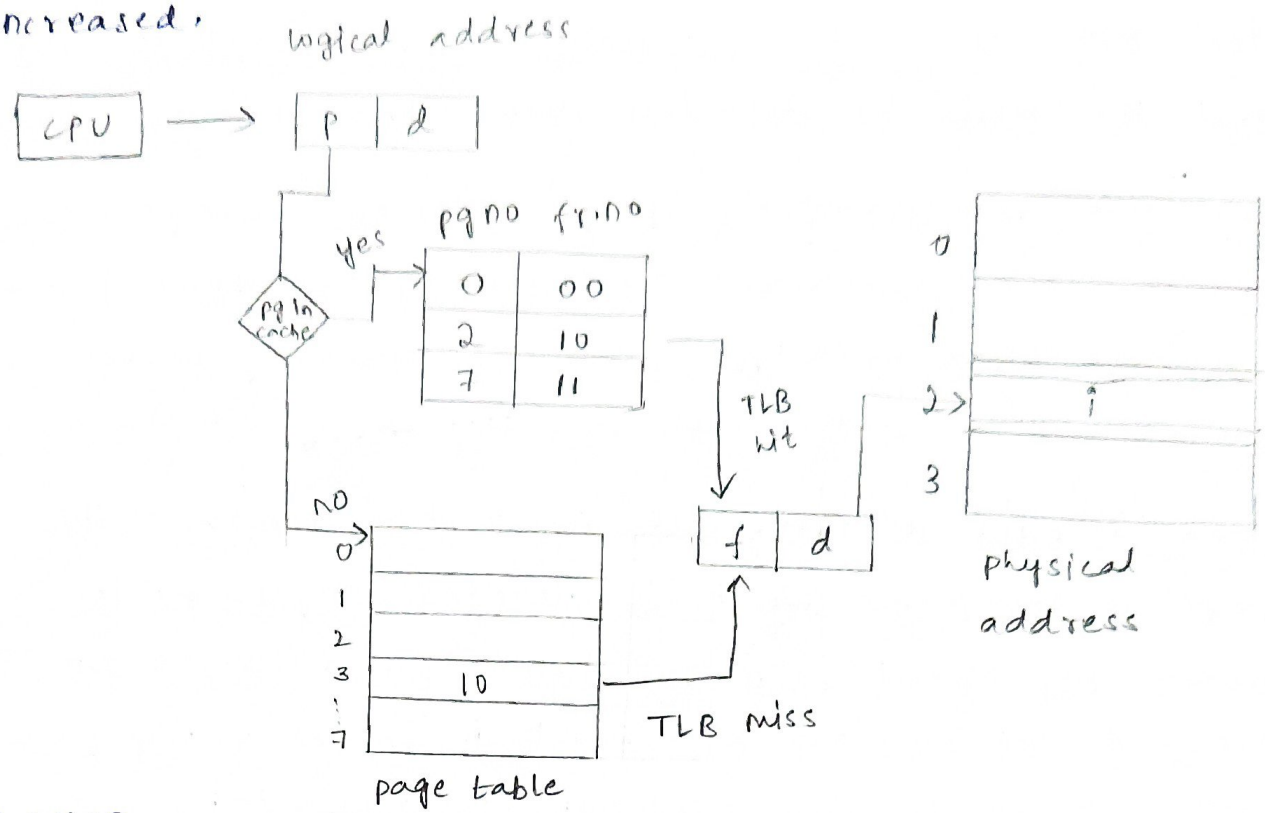


2) What is virtual memory? Explain the process of converting virtual addresses to physical addresses with a neat diagram.

A) Virtual memory is a storage scheme that provides user an illusion of having a very big main main. This is done by treating a part of secondary memory as the main memory.

Instead of loading one big process in the main memory, the OS loads the different parts of more than one process in main memory. By doing this the degree of multiprogramming will be increased and therefore, the CPU utilization will also be

increased.



⇒ When a program references a virtual address the MMU checks the TLB for a translaetion. If the translation is found, MMU directly uses the corresponding physical address. If the translation is not present in TLB, a page fault occurs. MMU triggers an interrupt and os takes over. The os retrieves the required page from secondary storage and brings into physical memory.

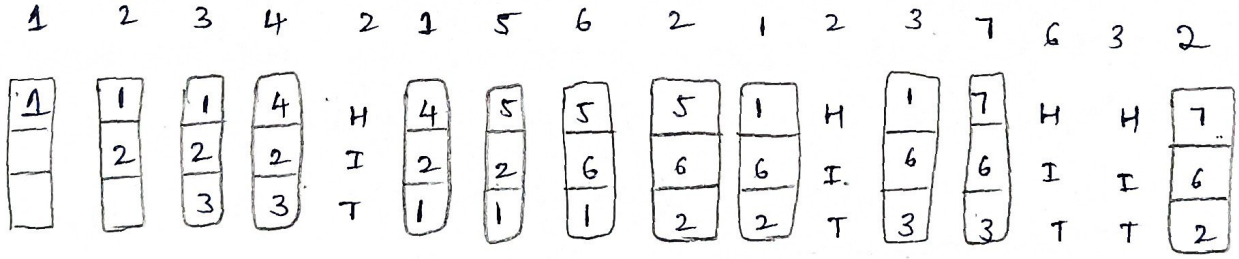
⇒ Once the page is loaded into physical memory, the os updates the page table to reflect the new mapping between the virtual page and the physical page.

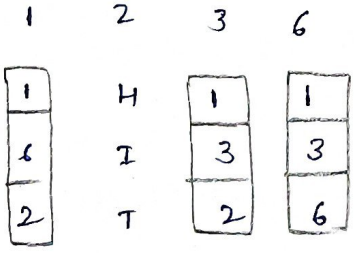
⇒ After handling the page fault, the memory management unit retrieves the virtual-to physical addresse translation.

This time the translation will be found in the TLB, and the MMU can directly access the physical memory

3) Consider the following page reference string: 1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6. Trace FIFO, optimal, LRU page replacement algorithms by assuming three frames and all frames are initially empty.

A) FIFO:- First in First out Algorithm, removes the page in the frame which is allotted long back. The useless page which is in the frame for a longer time is removed and the new page which is in the ready queue and is ready to occupy the frame is allowed by the FIFO page replacement.





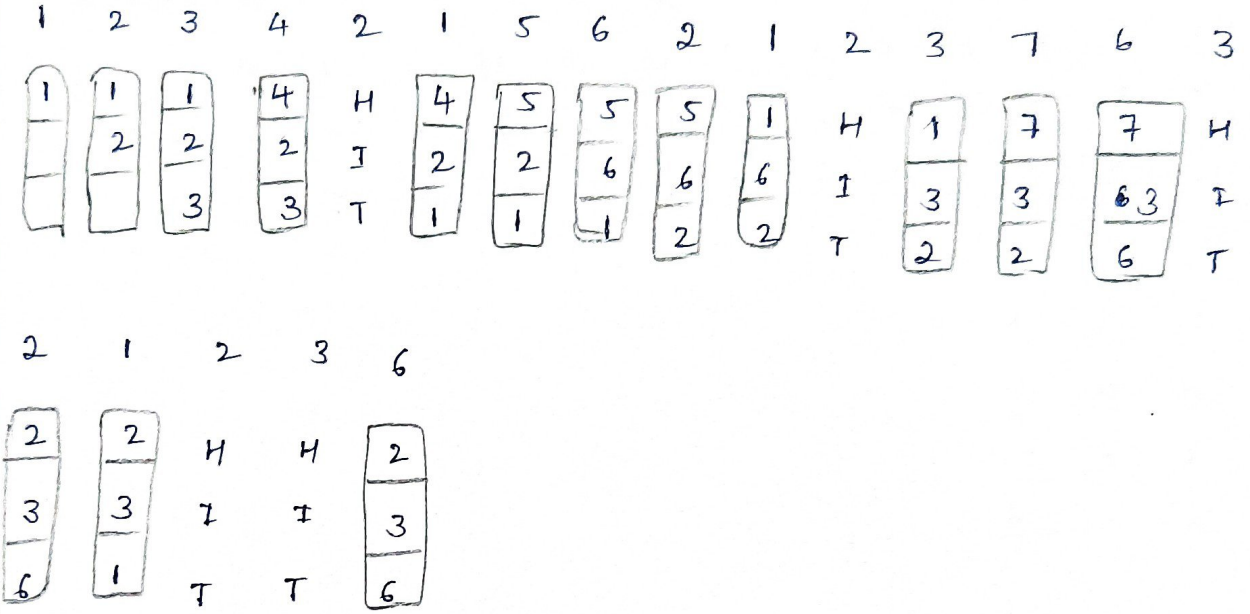
page reference string :- 20

page frame : 3

page hits : 5

page faults : 15

LRU:- Least Recently Used (LRU) page replacement Algorithm works on the principle - replace the page with the page which is less dimension of time recently used page in the past.



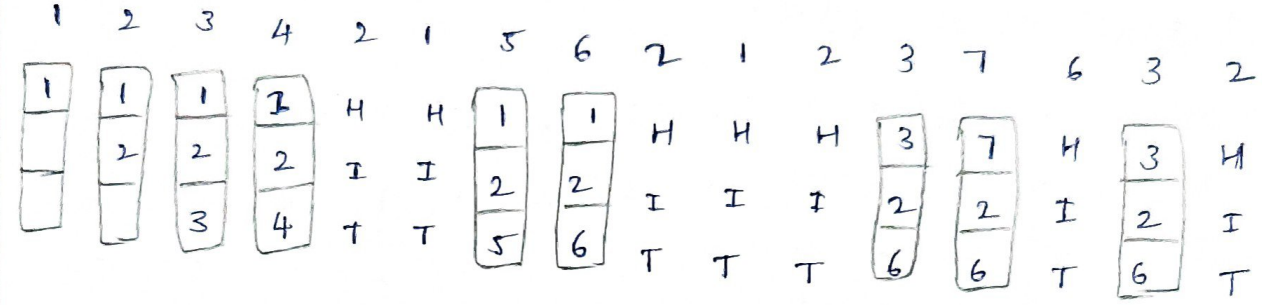
page reference string :- 20

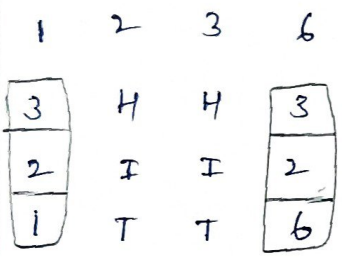
page frame : 3

page hits : 5

page faults : 15

optimal:- Optimal page replacement Algorithm works on the principle - Replace the page which is not used in the Longest dimension of time in future.





page reference string : 20

page frame : 3

page hits : 9

page faults : 11