



Virtual Memory – Page Replacement Algorithm

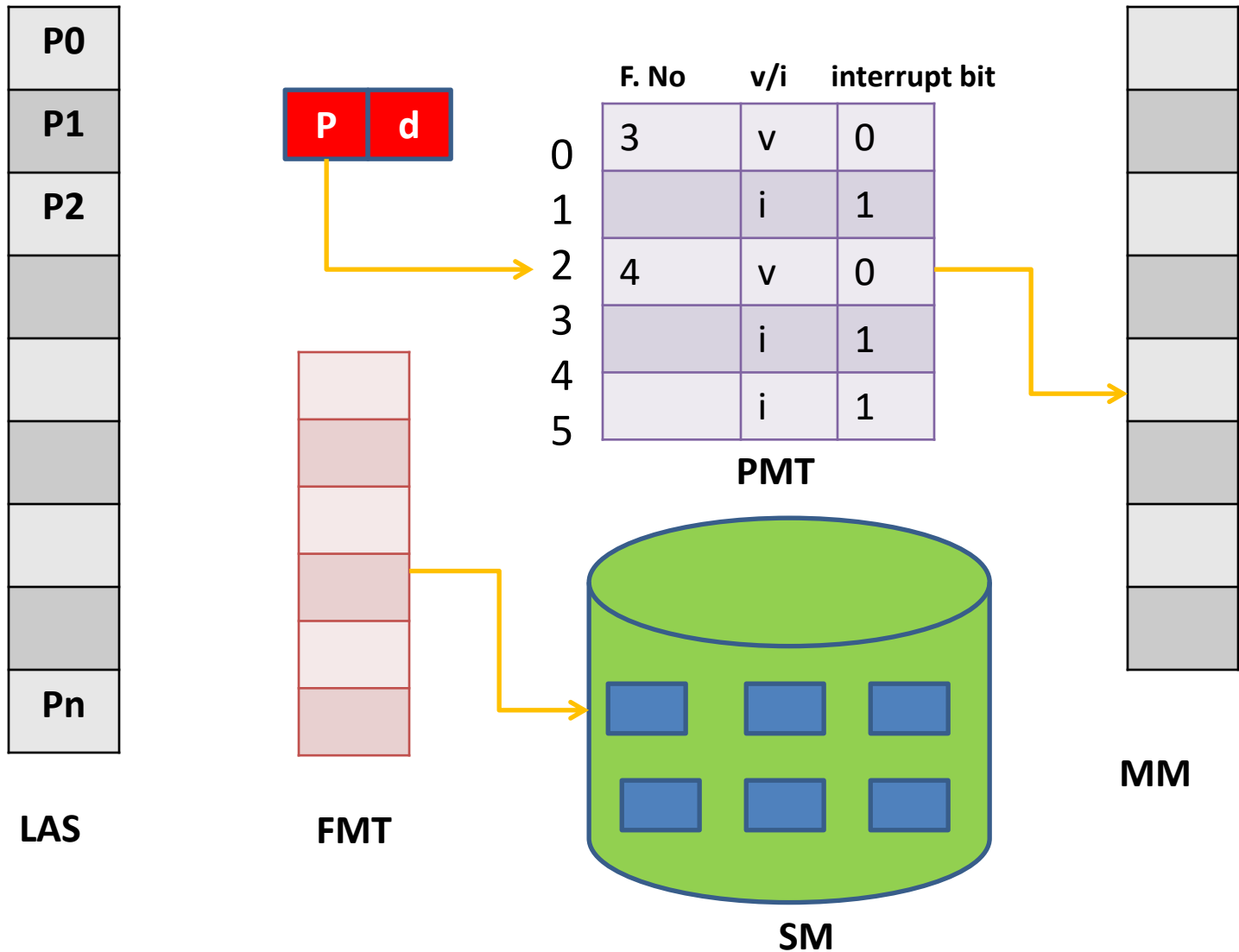
By

Dr. Upasana Pandey

Department of Computer Science & Engineering

IMS Engineering College (College Code:143)

Recap



H/W Support for Demand Paging

Recap

- A page fault occurs when a page referenced by the CPU is not found in the main memory.
- The required page has to be brought from the secondary memory into the main memory.
- A page has to be replaced if all the frames of main memory are already occupied.

Page Replacement:- Page replacement is a process of swapping out an existing page from the frame of a main memory and replacing it with the required page.

Page replacement is required when-

- All the frames of main memory are already occupied.
- Thus, a page has to be replaced to create a room for the required page.

Page Replacement Algorithm

Page Replacement Algorithms-

- Page replacement algorithms help to decide which page must be swapped out from the main memory to create a room for the incoming page.
- Various page replacement algorithms are-
 - First In First Out
 - Least Recently Used
 - Optimal Page Replacement

First In First Out (FIFO)

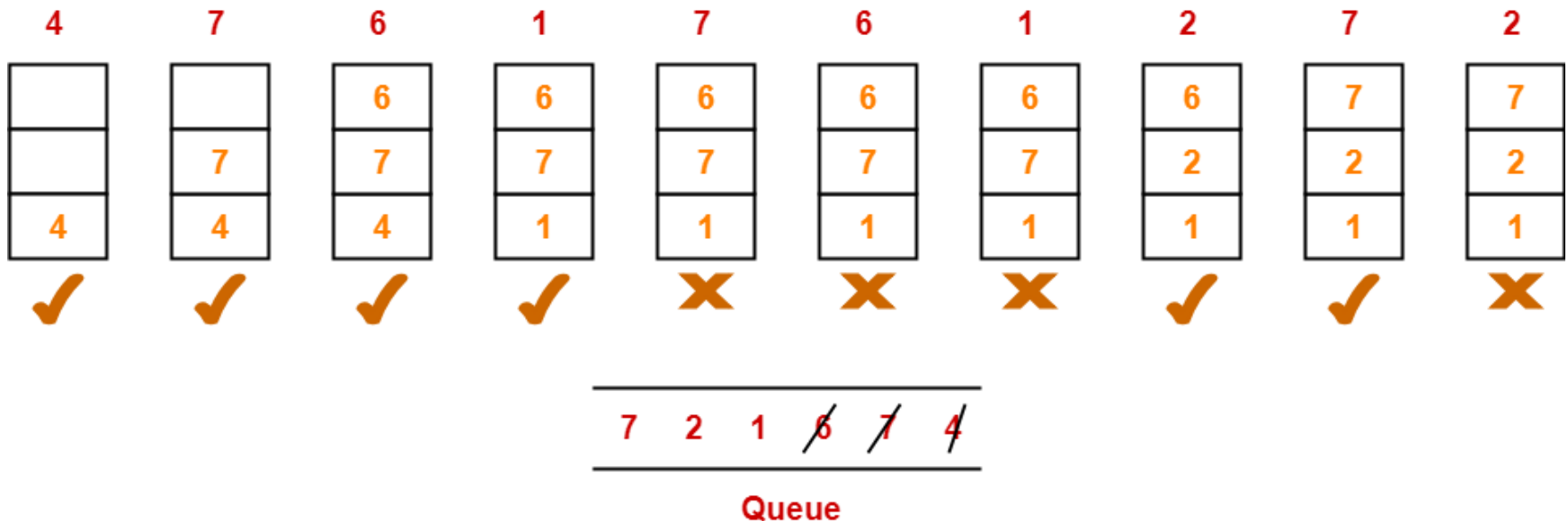
- As the name suggests, this algorithm works on the principle of “**First in First out**”.
- It replaces the oldest page that has been present in the main memory for the longest time.
- It is implemented by keeping track of all the pages in a queue.
- It is easy to understand and program.
- Its performance is not always good.
- It is not the first choice of OS.

First In First Out (FIFO)

A system uses 3 page frames for storing process pages in main memory. It uses the First in First out (FIFO) page replacement policy. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string given below-

4 , 7, 6, 1, 7, 6, 1, 2, 7, 2

Also calculate the hit ratio and miss ratio



First In First Out (FIFO)

Total number of page faults occurred = 6

Calculating Hit ratio-

Total number of page hits

= Total number of references – Total number of page misses or page faults

= 10 – 6

= 4

Thus, Hit ratio

= Total number of page hits / Total number of references

= 4 / 10

= 0.4 or 40%

Calculating Miss ratio-

Total number of page misses or page faults = 6

Thus, Miss ratio

= Total number of page misses / Total number of references

= 6 / 10

= 0.6 or 60%

Belady's Anomaly

Effect of Increasing Number of Frames-

- The number of page faults should either decrease or remain constant on increasing the number of frames in main memory.
- But sometimes the unusual behavior is observed.
- Sometimes, on increasing the number of frames in main memory, the number of page faults also increase.

Belady's Anomaly-

Belady's Anomaly is the phenomenon of increasing the number of page faults on increasing the number of frames in main memory.

Belady's Anomaly

FIFO page replacement algorithms suffer from Belady's Anomaly.

Important:

“Algorithms suffer from Belady's Anomaly” does not mean that always the number of page faults will increase on increasing the number of frames in main memory. This unusual behaviour is observed only sometimes.

Solution:

Stack based algorithms do not suffer from Belady's Anomaly. This is because these algorithms assign priority to a page for replacement that is independent of the number of frames in the main memory.

Following page replacement algorithms are stack based algorithms-

- **LRU Page Replacement Algorithm**
- **Optimal Page Replacement Algorithm**

Hence, they do not suffer from Belady's Anomaly.

Least Recently Used

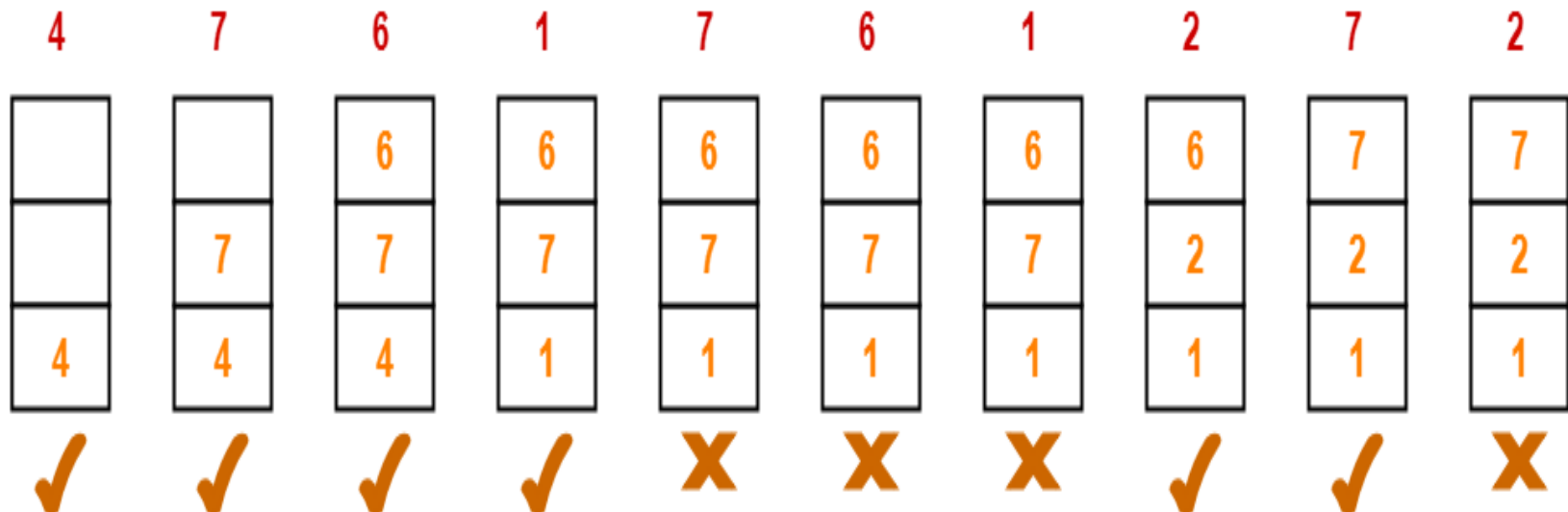
- It replaces the page in memory that has not been referenced for the longest time.
- It performs better than FIFO.
- It belongs to stack replacement algorithm
- Therefore does not suffer from Belady's anomaly.
- Its implementation imposes too much overhead to be handled.
- Stack is one of the solution for implementing LRU algorithm.
- Whenever a page is referenced, it is removed from the stack and put on the top.
- In this way, the top of the stack is always most recently used page and the bottom is LRU page.
- Implemented by doubly linked list.

Least Recently Used

A system uses 3 page frames for storing process pages in main memory. It uses the Least Recently Used (LRU) page replacement policy. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string given below-

4, 7, 6, 1, 7, 6, 1, 2, 7, 2

Also calculate the hit ratio and miss ratio.



Least Recently Used

Solution-

Total number of references = 10

Total number of page faults occurred = 6

In the similar manner as above-

- Hit ratio = 0.4 or 40%
- Miss ratio = 0.6 or 60%

Optimal Page Replacement

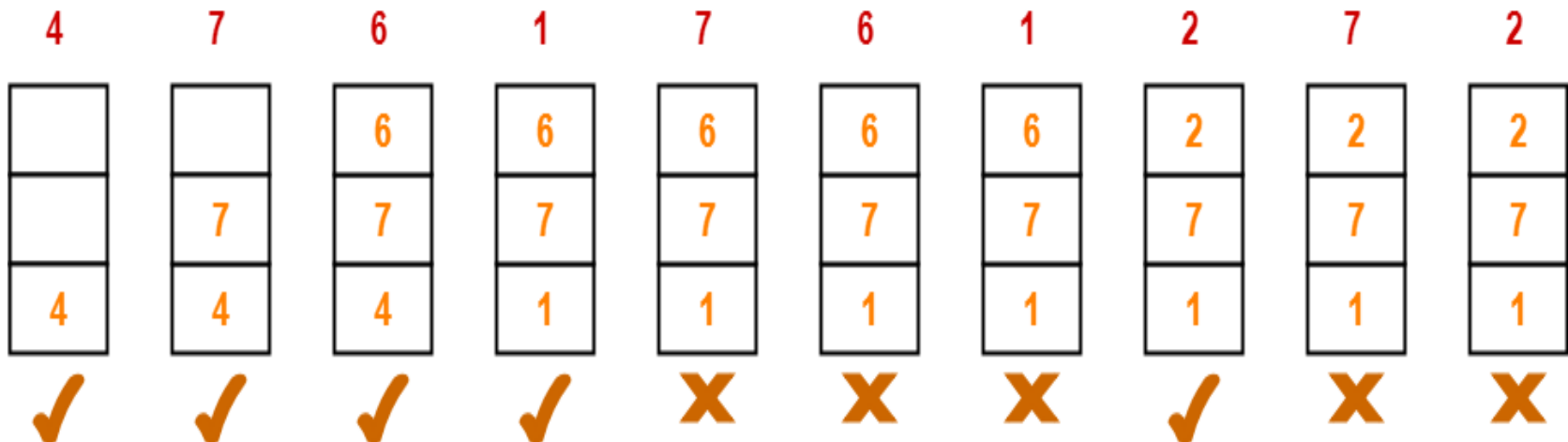
- This algorithm replaces the page that will not be referred by the CPU in future for the longest time.
- It is practically impossible to implement this algorithm.
- This is because the pages that will not be used in future for the longest time can not be predicted.
- However, it is the best known algorithm and gives the least number of page faults.
- Hence, it is used as a performance measure criterion for other algorithms.

Optimal Page Replacement

A system uses 3 page frames for storing process pages in main memory. It uses the Optimal page replacement policy. Assume that all the page frames are initially empty. What is the total number of page faults that will occur while processing the page reference string given below-

4 , 7, 6, 1, 7, 6, 1, 2, 7, 2

Also calculate the hit ratio and miss ratio.



Optimal Page Replacement

Total number of page faults occurred = 5

In the similar manner as above-

Hit ratio = 0.5 or 50%

Miss ratio = 0.5 or 50%

Thank You