**Page Replacement Schemes**

**Paging File**



**When RAM runs low, virtual memory can move data from it to a space called a paging file. This process allows for RAM to be freed up so that a computer can complete the task.**

**The file saves the data contained in the pages; if the program needs it again, the operating system reloads it when RAM becomes available.**



**Page replacement schemes**

**The memory management software system handles all the software operations for efficient utilization of memory space. This system decides**

**1. Which page in main memory to be removed to make a space for new page.**

**2. When a new page is transferred to main memory from secondary memory.**



**3. Where the page is to be placed in main memory.**

**When a page referenced by the CPU is not found in the main memory, it is called as a page fault.**

**When a page fault occurs, the required page is transferred from the secondary memory to main memory. If main memory is full, it would be necessary to remove a page from main memory block to make space for the new page. *The policy for choosing the pages to be removed is determined by using page replacement schemes.***



**The goal of replacement policy is trying to remove the page which is least likely to be referenced in the immediate future.**

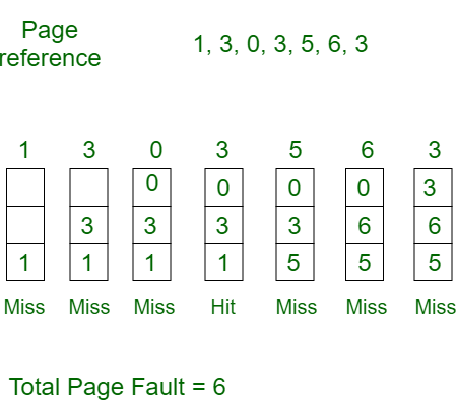
**Two of the most common replacement schemes used are *First-in first-out* (FIFO) and *the least recently used* (LRU).**

**The FIFO scheme selects for replacement of the page that has been in memory for the longest time.**

**First In First Out (FIFO) –**

**This is the simplest page replacement scheme. In this scheme, the operating system keeps track of all pages in the memory in a queue, the oldest page is in the front of the queue. When a page needs to be replaced page in the front of the queue is selected for removal. It means, in FIFO page replacement, when a page is needed to be replaced, we select the oldest page.**

**Example-1 Consider page reference string 1, 3, 0, 3, 5, 6 with 3-page frames. Find number of page faults.**

****

**Initially all slots are empty, so when 1, 3, 0 came they are allocated to the empty slots —> 3 Page Faults.**

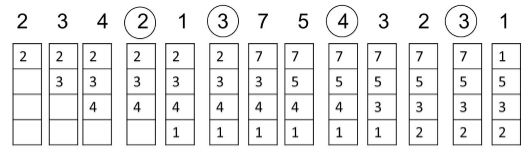
**when 3 comes, it is already in memory so —> 0 Page Faults.**

**Then 5 comes, it is not available in memory so it replaces the oldest page slot i.e 1. —>1 Page Fault.**

**6 comes, it is also not available in memory so it replaces the oldest page slot i.e 3 —>1 Page Fault.**

**Finally when 3 come it is not available so it replaces 0 🡪 1 page fault**

**FIFO page replacement scheme Example 2**

****

**Page hit: If the file is already present, then it is a Page Hit (indicated by circles in the diagram)**

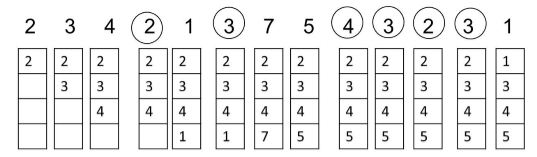
**Page Miss: If an entry is not found, then it is a Page miss**

**No. of page hit=4**

**No. of page miss=9**

**Optimal Page replacement:**

**Here, when a page replacement is needed, it looks ahead in the input queue for the page frame which will be referenced only after a long time. The page with the longest reference is swapped.**

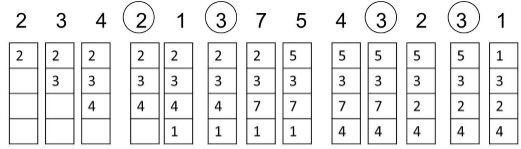
****

**No. of page hit=6**

**No. of page miss=7**

**LRU page replacement:**

**This method uses the recent past as an approximation of near future. We replace the page which has not been referenced for a long time in the past.**

****

**No. of page hit=4**

**No. of page fault=9**