TECHNO INTERNATIONAL BATANAGAR

TECHNO INTERNATIONAL BATANAGAR

Term Paper Report

FIFO Page Replacement Algorithm

NAME: SUJAN KUMAR MITRA

STREAM: CSE

YEAR: 2020

SEMESTER: 5TH

ROLL NO.: 33200118009

REG NO.: 183320110038

SUBJECT: Operating Systems

SUBJECT CODE: PCC-CS 502

FIFO Page Replacement Algorithm

Introduction:

In an operating system, we need to allocate memory to process to store the information regarding the process. Memory is allocated in the **Main Memory** primarily for faster access and suspended processes are stored in secondary memory.

One of the memory allocation strategies is called **Paging,** in which the process is divided into multiple pages, the main memory is also divided into multiple frames, and these pages are stored in frames of memory (page size must be equal to frame size). The **MMU** (Memory Management Unit) is responsible for mapping the pages of a process to frame address in which the page is kept. This technique, eliminates the **Internal Fragmentation**, and helps in reducing wastage of memory.

In a multiprogramming operating system, the operating system tries to maximize the **degree of multiprogramming** by keeping as many processes as possible in the main memory. But, since the capacity of main memory is limited, so we need to design some strategies so that, we can keep only the necessary pages of a process in the main memory and rest pages are swapped out in secondary/virtual memory.

When CPU demands for a page of a process, the MMU returns the page address, if it is present in main memory, or we bring the page from secondary memory to primary memory. If there is no more space available in main memory to allocate, then it replaces an existing page from main memory to secondary memory.

When MMU performs the swap operation, if a page is not found in main memory, it is called a **Page Fault**. Page Faults are costly, because swapping pages results in a context switch, control goes from Process to OS then again comes back to Process. So, we need to design algorithms, such that, occurrence of page faults is minimum. FIFO Page Replacement Algorithm is one such algorithm.

FIFO Page Replacement Algorithm:

In this algorithm, the operating system replaces the oldest page present in the memory. This is a simple algorithm can be easily implemented with a queue. The oldest page is kept at the front of the queue. When a page needs to be replaced page at the front of the queue is selected for removal.

Algorithm:

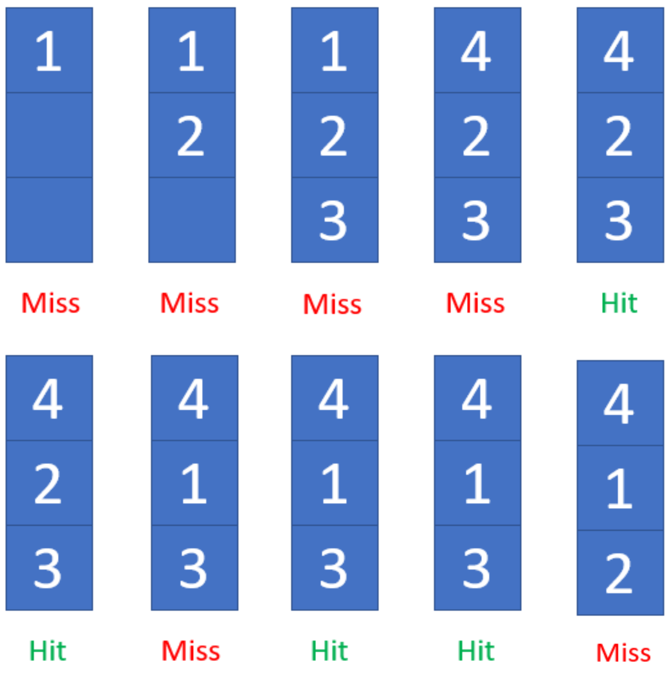
```
i) If set holds less pages than capacity.
   a) Insert page into the set one by one until
      the size of set reaches capacity or all
      page requests are processed.
   b) Simultaneously maintain the pages in the
      queue to perform FIFO.
ii) Else
   If current page is present in set, do nothing.
   Else
     a) Remove the first page from the queue
        as it was the first to be entered in
        the memory
     b) Replace the first page in the queue with
        the current page in the string.
     c) Store current page in the queue.
```

Example:

Consider a situation where CPU demand pages in the following order, calculate the page fault rate, for frame capacity 3 and 4.

Order: **1, 2, 3, 4 ,2, 3, 1, 3, 4, 2**

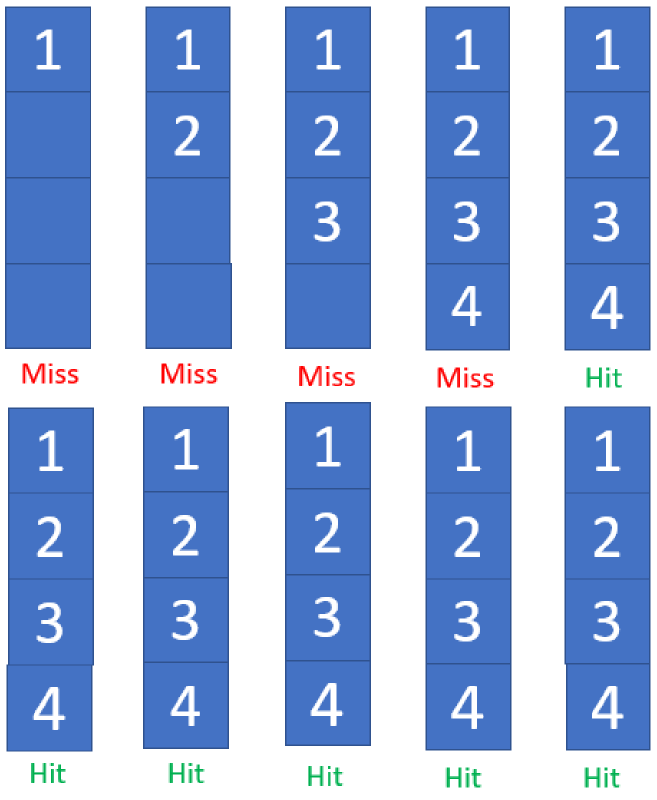
Frame Capacity=3



Page Fault Rate:

60%

Frame Capacity=4



Page Fault Rate:

40%

Belady’s Anomaly:

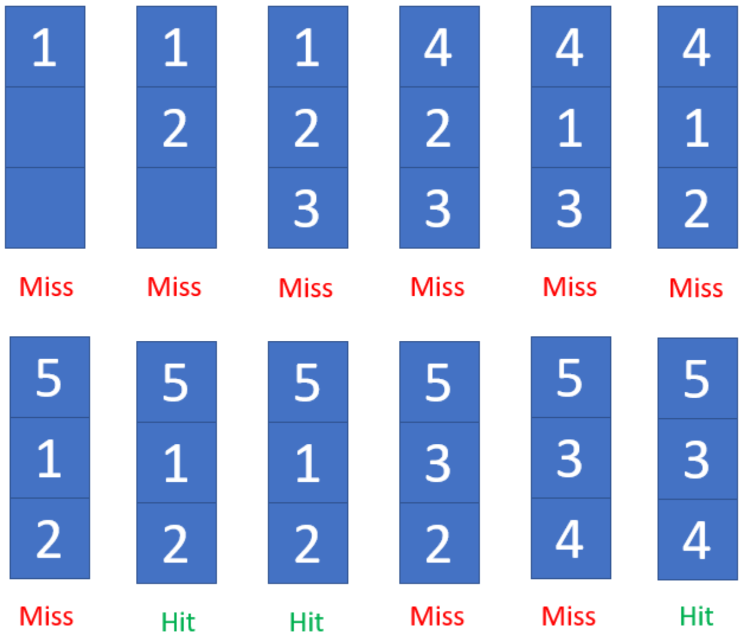
Generally, in page replacement algorithms, if the capacity of frames is increased, then the page fault rate is expected to reduce like we saw in previous example. But in exceptional cases, on increasing the frame capacity, the page fault rate is increased. This anomaly is demonstrated by Bélády, hence called Belady’s Anomaly.

Example:

Consider the following page reference:

1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5

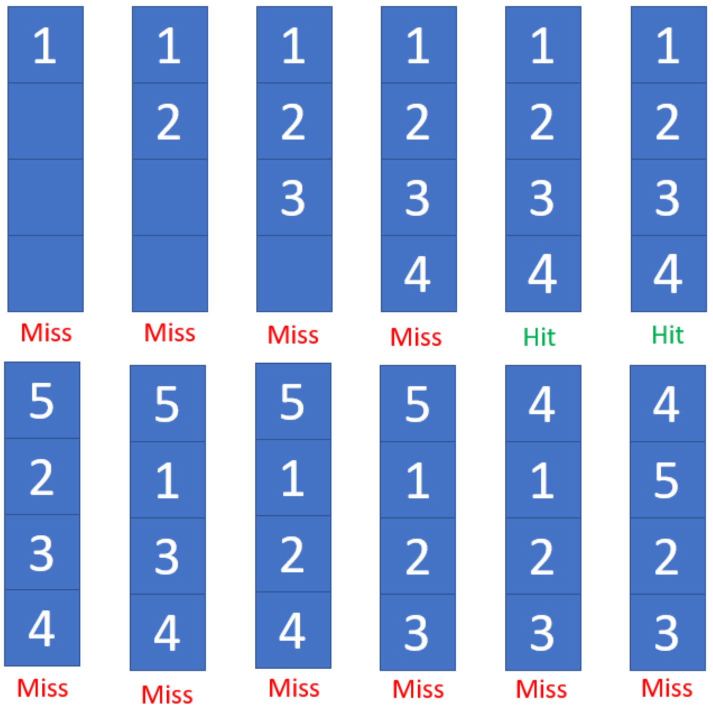
Frame Capacity=3:



Page Fault Rate:

75%

Frame Capacity=4:



Page Fault Rate:

83.33%

Advantages and Disadvantages:

Advantages:

- It is simple and easy to understand & implement.

Disadvantages:

- The process effectiveness is low.
- Belady’s Anomaly.
- Every frame needs to be taken account off.