International Journal of Advances in Engineering and Management (IJAEM)
Volume 2, Issue 10, pp: 148-150 www.ijaem.net ISSN: 2395-5252

An Overview of Page Replacement Algorithms

Adarsh.S, Vineeth.J, Sacchin Adarsh.V.S, Dr.M. Sujithra M.C.A, M.Phil., PhD, Dr.A.D. Chitra M.C.A, M.Phil., PhD,

II-Year, M.Sc. Software Systems (Integrated), Coimbatore Institute of Technology.
Assistant Professor, Department of Data Science, Coimbatore Institute of Technology, Coimbatore
Assistant Professor, Department of Software Systems, Coimbatore Institute of Technology, Coimbatore

Date of Submission: 20-11-2020 Date of Acceptance: 06-12-2020

ABSTRACT: This paper is about Page Replacement Algorithms. In a computer operating system that uses paging for virtual memory management, page replacement algorithms decide which memory page to be loaded into the main memory (swap in, write to RAM) when a page of memory needs to be allocated.A virtual memory system requires efficient page replacement algorithms to make a decision which pages to evict from memory in case of a page fault. Many algorithms have been proposed for page replacement. Each algorithm is used to decide on which free page frame, a page is placed and tries to minimize the page fault rate while incurring minimum overhead.

KEYWORDS: Virtual memory, page fault, page hit, page replacement algorithms.

I. INTRODUCTION:

Whenever a process is to be executed, the process has to be brought into the main memory (RAM) and then processed.The processes are brought into the RAM as and only when required.Pages are brought into main memory onlywhen the executing process demands them, this is known as demand aging.Since the RAM has fixed amount of memory space, OS makes use of the virtual memory. Virtual Memory is a storage allocation scheme used by the Memory Management Unit(MMU) to compensate for the shortage of physical memory by transferring data from RAM to disk storage.A page fault typically occurs when a process references to a page that is not marked present in main memory and needs to be broughtfrom secondary memory. In such a case an existing page needs to be discarded. The selection of such a page is performed by pagereplacement algorithms.

When an executing process refers to a page, it is first searched in the main memory. If it is not present in the main memory, a page fault occurs.**Page Fault** is the condition in which a running process refers to a page that is not loaded in the main memory. If the page is already present in the memory, it is known as **Page Hit.** And the process of replacing the pages between main memory and virtual memory is called **Swapping.**

Hit ratio = Total number of Page Hits / Total number ofReference Counts.
Fault ratio = Total number of Page Faults / Total number of Reference Counts.

PAGE REPLACEMENT ALGORITHMS:

Page Replacement Algorithm decides which page to remove, also called swap out when a new page needs to be loaded into the main memory. Page Replacement happens when a requested page is not present in the main memory and the available space is not sufficient for allocation to the requested page. When the page that was selected for replacement was paged out, and referenced again, it has to read in from disk, and this requires for I/O completion. This process determines the quality of the page replacement algorithm: the lesser the time waiting for page-ins, the better is the algorithm.

A page replacement algorithm tries to select which pages should be replaced so as to minimize the total number of page misses. There are many different page replacement algorithms. These algorithms are evaluated by running them on a particular string of memory reference and computing the number of page faults.The fewer is the page faults the better is the algorithm for that situation.

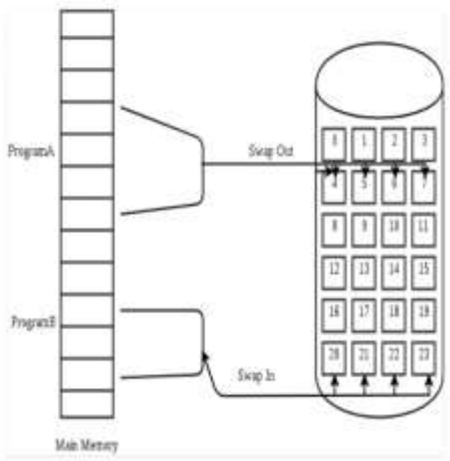
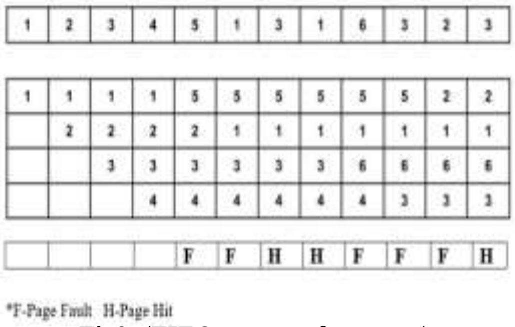


Fig1.(Page replacement-swapping)

FIFO (First In First Out):

This is the simplest page replacement algorithm. In this algorithm, the OS maintains a queue that keeps track of all the pages in memory, with the oldest page at the front and the most recent page at the back.When there is a need for page replacement, the FIFO algorithm, swaps out the page at the front of the queue, that is the page which has been in the memory for the longest time.

Consider the page reference string of size 12**: 1, 2, 3, 4, 5, 1, 3, 1, 6, 3, 2, 3** with frame size 4



1	2	3	4	5	1	3	1	6	3	2	3

1	1	1	1	5	5	5	5	5	5	2	2
	2	2	2	2	1	1	1	1	1	1	1
		3	3	3	3	3	3	6	6	6	6
			4	4	4	4	4	4	3	3	3

				F	F	H	H	F	F	F	H

*F-Page Fault H-Page Hit

Fig2. (FIFO page replacement)

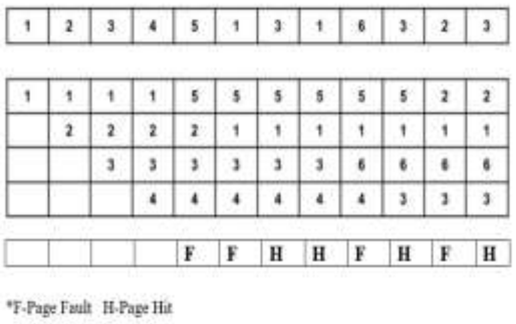
ADVANTAGES AND DISADVANTAGES OF FIFO:

- Simple and easy to implement.
- Low overhead.
- Poor performance.
- Doesn't consider the frequency of use or last used time, simply replaces the oldest page.
- Suffers from Belady's Anomaly(i.e. more page faults when we increase the number of page frames).

LRU (Least Recently Used):

Least Recently Used page replacement algorithm keeps track of page usage over a short period of time. It works on the idea that the pages that have been most heavily used in the past are most likely to be used heavily in the future too.In LRU, whenever page replacement happens, the page which has not been used for the longest amount of time is replaced.

Consider the page reference string of size 12**: 1, 2, 3, 4, 5, 1, 3, 1, 6, 3, 2, 3** with frame size 4.



1	2	3	4	5	1	3	1	6	3	2	3

1	1	1	1	5	5	5	5	5	5	2	2
	2	2	2	2	1	1	1	1	1	1	1
		3	3	3	3	3	3	6	6	6	6
			4	4	4	4	4	4	3	3	3

				F	F	H	H	F	H	F	H

*F-Page Fault H-Page Hit

Fig3. (LRU page replacement)

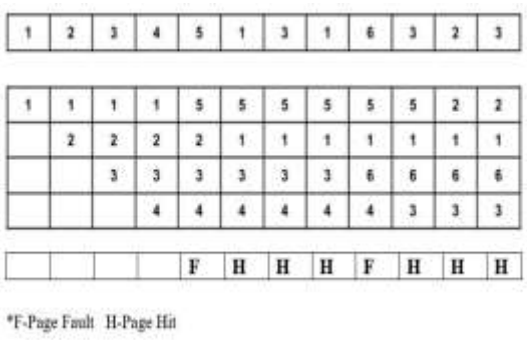
ADVANTAGES AND DISADVANTAGES OF LRU:

- Efficient.
- Doesn't suffer from Belady's Anomaly.
- Complex Implementation.
- Expensive.
- Requires hardware support.

OPTIMAL PAGE REPLACEMENT:

Optimal Page Replacement algorithm is the best page replacement algorithm as it gives the least number of page faults. It is also known as OPT, clairvoyant replacement algorithm, or Belady's optimal page replacement policy.In this algorithm, pages are replaced which would not be used for the longest duration of time in the future, i.e., the pages in the memory which are going to be referred farthest in the future are replaced. This algorithm was introduced long back and is difficult to implement because it requires future knowledge of the program behaviour. However, it is possible to implement optimal page replacement on the second run by using the page reference information collected on the first run.

Consider the page reference string of size 12**: 1, 2, 3, 4, 5, 1, 3, 1, 6, 3, 2, 3** with frame size 4.



1	2	3	4	5	1	3	1	6	3	2	3

1	1	1	1	5	5	5	5	5	5	2	2
	2	2	2	2	1	1	1	1	1	1	1
		3	3	3	3	3	3	6	6	6	6
			4	4	4	4	4	4	3	3	3

				F	H	H	H	F	H	H	H

*F-Page Fault H-Page Hit

Fig4. (Optimal Page Replacement)

ADVANTAGES AND DISADVANTAGES OF OPTIMAL:

- Easy to Implement.
- Simple data structures are used.
- Highly efficient.
- Requires future knowledge of the program.
- Time-consuming.

II. ANALYSIS AND CONCLUSION:

Number of page faults and their ratio:

FIFO	:	5 (9/12)
LRU	:	4 (8/12)
OPTIMAL	:	2 (6/12)

The purpose of this work was to define the algorithm that realizes the best performance of the system. A good page replacement algorithm can reduce the page faults, when the program is executing, reduce the number of I/O, and then increase the system's efficiency effectively. The time is a critical point for the systems. An improvement in the performance of the system, having less page faults is made. FIFO has the worst performance. It has more page faults (degenerates) when the number of pages is increased. Optimal is the best algorithm.

REFERENCES:

[1]. William Stallings, "Operating Systems Internals and Design Principles" , Ninth Edition
[2]. https://afteracademy.com/blog/what-are-the-page-replacement-algorithms.
[3]. Juhi Kumari, Sonam Kumari & Devendra Prasad, A Comparison of Page Replacement Algorithms: A Survey International Journal of Scientific & Engineering Research, Volume 7, Issue 12, December-2016.