31118

PICT, PUNE

Subject: Laboratory Practices-1

Assigmoment No: 03

Date: 21-Sept-2021

Title: Page Replacement

Problem statement Write a program to simulate page replacement algorithm

S/W & h/w requirement

Hardware:

Manufactures: Acer Inc.

Process: Intel core i5-8265U @1.60GHz

Ram: 8GB, ROM: 512GB SSD

Software:

Operating System: windows-10 (64-bit)

~~IDE~~ Code editor: Sublime Text

Compiler: GCC version 6.3.0.

Theory:

Page Replacement:

1) Page replacement is used in memory management.

2) A computer system has limited amount of memory. Adding more memory using combination of both hardware & software to allow the computer to address more memory than physically present. This extra memory is called virtual memory.

3) Virtual memory can be implemented using two methods:

i) Paging

ii) Segmentation.

4) Paging is a process of reading data from & writing

data to, secondary storage
The main objective of paging is to divide each process in form of pages of fixed size.
Pages of a process are only brought from secondary memory to main memory when they are needed

5) When an executing process refers to a pages it is first searched in main memory. If it is not present there, page fault occurs.
In such case, os bring the page from secondary memory into main memory, this may cause some pages in main memory to be replaced. ~~by~~
For optimal straterglies page replacement algorithms are used.

6) Page replacement Algorithm decide which page to remove, when new page is added
Page replacement occurs when requested page is not present in main memory & available space is not sufficient for allocation of requested page

Page Replacement Algorithms:

A) First in First Out (FIFO)

i) In this algorithm, the os maintains a queue that keep track of all pages in memory, with oldest page at front & most recent at back

ii) When there is need for page replacement, it swap out the front page of queue

B> <u>Least Recently Used (LRU)</u>

i> This algorithm keeps track of page usage over a short period of time.

ii> In LRU, whenever a page replacement happen, the page which has not been used for longest amount of time is replaced.

C> <u>Optimal Page Replacement</u>

i> In this algorithm, pages are replaced which would not be used for the longest duration of time in the future.

ii> It requires process sequence details in advance.

Q> Advantages & disadvantages.

Algorithm	Advantages	Disadvantages
i> FIFO	i> simple & easy to ~~ii>~~ implement. ii> Low overhead	i> Poor performance ii> Suffers from Belady's Anomaly (i.e. more page faults when we increase number of page frames)
ii> LRU	i> Efficient ii> Doesn't suffer from Belady's Anomaly	i> Complex implementation ii> Expensive iii> Requires hardware support.

Optimal Page Replacement	i> Easy to implement ii> Simple data structures iii> Highly efficient.	i> Requres futue knowoledge of the program. ii> Time consuming

Algorithm/Flow chart:

FIFO:

```
1> start traversing the pages.
    i> If set holds less pages than capacity
        a> insert page into set
        b> increment page fault
    ii> else
        If current page is present in set, do nothing
        else
            a> remove first page from queue.
            b> store current page in queue.
            c> increment page fault
2> return page faults.
```

LRU:

```
1> Start traversing the pages.
    i> if set holds less pages than capacity
        a> inset page into set.
        b> increment page fault.
    ii> else
        If current page is present in set put it
            in front of queue.
        else remove back of queue & increment
            page fault count. insert current page at
            front of queue.
```

2) return page fauts

Optimal Page Replacement

```
1) Start traversing the pages
    i) if set holds less pages than capacity
        a) insert page into set
        b) increment page fault
    ii) else
        If current page is present in set. do nothing
        else remove the page from set which will
        be used last in future. Also increment page
        faults & put current page in set.
2) Return page faults.
```

Testcases:

Consider sequence of pages:

7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1

& capacity = 3.

fifo page replacement:

7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1

first come →	7	7	7	0	1	2	2	3	0	4	2	2	0	3	0	0	0	7	7	7
		0	0	1	2	0	3	0	4	2	3	0	3	0	1	1	1	1	0	0
			1	2	0	3	0	4	2	3	0	3	2	1	2	2	2	2	2	1

LRU Page Replacement:

7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1

	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
		7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0
LRU. →			7	0	1	2	2	3	0	4	2	2	0	3	3	1	2	0	1	7
	↑	↑	↑	↑		↑		↑	↑	↑	↑			↑		↑		↑		

Optimal Page Replacement:

7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1

7	7	7	2	2	2	2	2	2	2	2	2	2	2	2	2	2	7	7	7
	0	0	0	0	0	0	4	4	4	0	0	0	0	0	0	0	0	0	0
		1	1	1	3	3	3	3	3	3	3	3	1	1	1	1	1	1	1
↑	↑	↑	↑		↑		↑			↑			↑				↑		

Comparison:

Algorithm	Page faults	Page ~~Misses~~ hits
FIFO	15	5
LRU	12	8
optimal page replacement	9	11

Conclusion:

In this assignment, we learned about various page replacement algorithms. Also simulated them & done result analysis.