	DOP: 21 OC+,2021 31443.
313A.238	Assignment No: 07 (Group B): Page Replacement Algorithm.
1)	Jitle: Write a program to simulate page replace- ment algorithm.
2)	Software and hardware requirement:
	* Softwares)
	THE RESIDENCE OF THE PROPERTY
	1) Integraled Development Environment: - CLion.
	2) Cpp Compiler. 3) (OR) Notipad.
	3) (OK) Notipad.
	Hardware
1	
	1) Computer System:
	1) Computer System:- 2) processor: is gen
	ь) Ram: 8 GB
	c) Monitor: 1080p 1720p fHDIEPS.
	d) I/O Peripherals
	- keyboard
	- Mouse.
3)	Léauning Objedive:
	1) Understand what is page replacement Algorithm.
	Algorithm.



- 2) Understand what is demand paging.
  3) Understand page replacement algorithm and its working.
- 4) Leauring Outcome:
  - one will be able to understand the concept of page replacement its algorithms and what is demand paging.
  - 2) One will be able to implement frage replacement algorithms using appropriate clota structure set and programming & & 11 set.
- 5) Concept Related Theory

Demand Paging: According to the concept of Virtual memory, in order to execute some fuscess, only a hard of the process needs to be present in the main memory which means that only a new hages will only be present in the main memory at any time.

However, deciding, which page need to be kept in the main memory and which need to be kept in the secondary memory, is going to be difficult because we cannot say in advance that a pre-



will require a particular page at particular time

Therefore, to overcome this problem there is a concept called clemand paging is introduced it suggests being all pages of the frame in the secondary memory until they one required. In other words it says that do not load any page in the main memory until it is required.

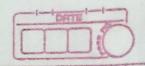
Whenever vary page is reserved for the first time in the main memory, then that page will be found in the secondary memory.

After that, it may are may not be present in the main memory depending upon the page replacement algorithm which will be covered later in this tutorial.

# \* Page Replacement Algorithm in 03.

In operating system that uses haging for memory management, a page replacement algorithm is needed to clecide which page needs to be replaced when new pages comes in.

Page fault: a page fault happons when a ruenning program access a memory page that is mapped into the virtual address space but mot hot loaded in physical memory.



Since, Actual memory is much smaller than virtual memory, page foult happens. In Cose of page facult, OS might have to replace one of the existing pages with the newly needed page. Different page replacement algorithms suggest different ways to decide which page to replace the target for all algorithms is to reduce the number of page faults.

### 1) first in-first-out [FIFO]:

The OS keeps track of all pages in the memory in a queue the oldse page is in the front of the Gueve. When a page needs to be suplaced page in the front of grene is selected for removal.

Example:

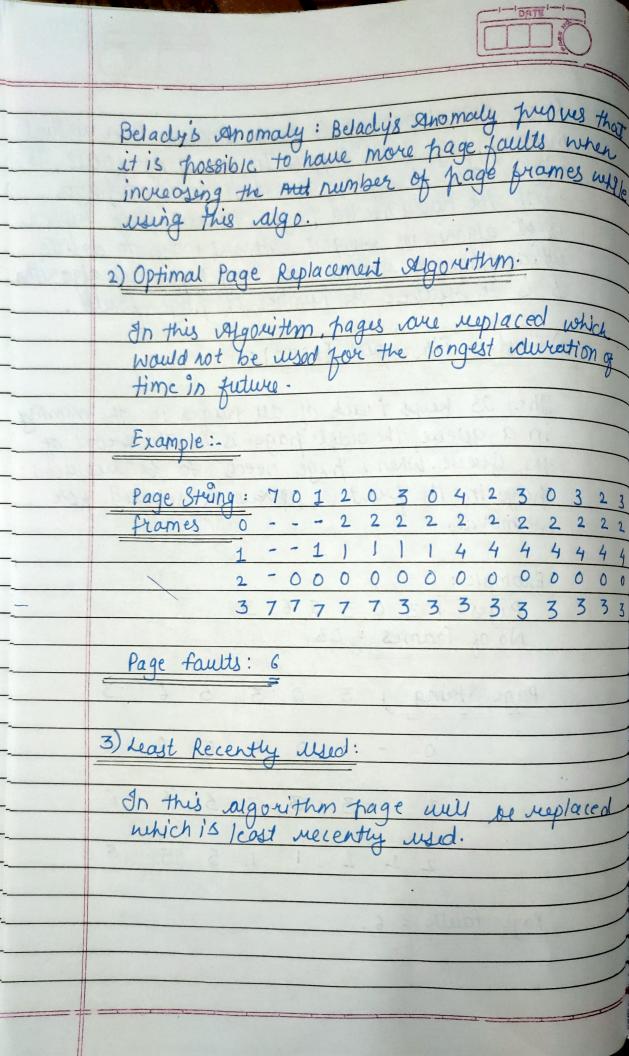
Pages: 1 3 0 3 5 6 3 No of Frames: 08. Page String 1 3 0 3

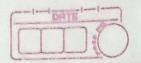
0 0 0

> 1 - 3 3 3 3 6 6

2 1 1 5 1 1 5 5

Page faults = 6.





Exam	D	le	
	-		

Page fauts: 6.

### 6) Conclusion:

understood and implemented the concept of page replacement Algorithms.

## 7) Reperences

- 1) Geeks For Geeks.
- 2) Jowa Tpoint.

#### Code:

```
#include<iostream>
int *initializeFrames(int frames[],int numOfFrames){
```

```
void algoLRU(int numOfPages,int numOfFrames,int pageString[],int frames[]){
   int n=numOfPages, nf=numOfFrames, i, pos=0;
```

```
initializeFrames(frames, numOfFrames);
```

```
if(cachePosPointer==numOfFrames) cachePosPointer = 0;
```

```
cin>>numOfPages;
cout<<endl<<"Enter number of Frames : ";</pre>
cin>>numOfFrames;
            cout << "\nPlease Enter valid choice between 1 to 4\n";</pre>
```

#### **Output:**

Enter number of Pages :14

Enter number of Frames :4

Enter Pages ->

Enter page 1:7

Enter page 2:0

Enter page 3:1

Enter page 4 :2

Enter page 5:0

Enter page 6:3

Enter page 7:0

Enter page 8:4

Enter page 9:2

Enter page 10:3

Enter page 11:0

Enter page 12:3

Enter page 13:2

Enter page 14:3

Enter your choice :1

<sup>\*\*</sup>Page Replacement using Least Recently Used Algorithm\*\*

Page 7 ->	7	-	-	-	'F'
Page 0 ->	7	0	-	-	'F'
Page 1 ->	7	0	1	-	'F'
Page 2 ->	7	0	1	2	'F'
Page 0 ->	7	0	1	2	'H'
Page 3 ->	3	0	1	2	'F'
Page 0 ->	3	0	1	2	'H'
Page 4 ->	3	0	4	2	'F'
Page 2 ->	3	0	4	2	'H'
Page 3 ->	3	0	4	2	'H'
Page 0 ->	3	0	4	2	'H'
Page 3 ->	3	0	4	2	'H'

Page 2 -> 3 0 4 2 'H' Page 3 -> 3 0 4 2 'H'

Page faults: 6 Page Hit: 8

Enter your choice :2

\*\*Page Replacement using Optimal Algorithm\*\*

'F' Page 7 -> 7 Page 0 -> 'F' 7 0 Page 1 -> 7 'F' 0 1 'F' Page 2 -> 7 0 1 2 Page 0 -> 7 0 1 2 'H' 'F' Page 3 -> 3 0 1 2 Page 0 -> 'H' 0 2 3 1 2 'F' Page 4 -> 3 0 4 Page 2 -> 3 0 4 2 'H' 2 Ή' Page 3 -> 3 0 4 Page 0 -> 3 0 4 2 'H' Page 3 -> 2 'H' 3 0 4 2 'H' Page 2 -> 3 0 4 'H' Page 3 -> 4 2

Page Faults: 6 Page Hits: 8

### Enter your choice :3

\*\*Page Replacement using FIFO Algorithm\*\*

Page 7 ->	7	-	-	-	'F'
Page 0 ->	7	0	-	-	'F'
Page 1 ->	7	0	1	-	'F'
Page 2 ->	7	0	1	2	'F'
Page 0 ->	7	0	1	2	'H'
Page 3 ->	3	0	1	2	'F'
Page 0 ->	3	0	1	2	'H'
Page 4 ->	3	4	1	2	'F'
Page 2 ->	3	4	1	2	'H'
Page 3 ->	3	4	1	2	'H'
Page 0 ->	3	4	0	2	'F'
Page 3 ->	3	4	0	2	'H'
Page 2 ->	3	4	0	2	'H'
Page 3 ->	3	4	0	2	'H'

Page Faults: 7 Page Hits: 7

Menu	١
1: Least Recently Used	l
2: Optimal	
3: First In First Out	
4: Exit	

Enter your choice :4

Terminated..!

Process finished with exit code 0