Mawlana Bhashani Science and Technology University

Lab-Report

Report No:11

Course code:ICT-3110

Course title:Operating Systems Lab

Date of Performance:11-09-2020

Date of Submission:16-09-2020

Submitted by

Name:Meraz Ahmed

ID:IT-18005

3rd year 1st semester

Session: 2017-2018

Dept. of ICT

MBSTU.

Submitted To

Nazrul Islam

Assistant Professor

Dept. of ICT

MBSTU.

Experiment No: 11

Experiment Name: Implementation of FIFO page replacement Algorithm

Aim and Objects:

It is one of the simplest page replacement algorithm. The oldest page, which has spent the longest time in memory is chosen and replaced. This algorithm is implemented with the help of FIFO queue to hold the pages in memory. A page is inserted at the rear end of the queue and is replaced at the front of the queue

Page Replacement Algorithms :

- **First In First Out (FIFO) –**
-
  This is the simplest page replacement algorithm. In this algorithm, 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.

  - **east Recently Used –**
    In this algorithm page will be replaced which is least recently used.

  - **Example-3** Consider the page reference string 7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2 with 4 page frames.Find number of page faults.

Code:

```c
#include<stdio.h>
int main()
{
    int i,j,n,no,k,avail=0,count=0,value[50],frame[10];
    printf("\n Enter the number of pages  ");
    scanf("%d",&n);
    printf("\n Enter the pages :\n");
    for(i=1; i<=n; i++)
        scanf("%d",&value[i]);
    printf("\n Enter the number of frame : ");
    scanf("%d",&no);
    for(i=0; i<no; i++)
        frame[i]= -1;
```

```c
    j=0;
    printf("Ref string\t page frames\n");
    for(i=1; i<=n; i++)
    {
        printf("%d\t",value[i]);
        avail=0;
        for(k=0; k<no; k++)
            if(frame[k]==value[i])
                avail=1;
        if (avail==0)
        {
            frame[j]=value[i];
            j=(j+1)%no;
            count++;
            for(k=0; k<no; k++)
                printf("%d\t",frame[k]);
        }
        printf("\n");
    }
    printf("\nPage Fault Is %d\n",count);
    return 0;
}
```

Output:

```
 Enter the number of pages  10

 Enter the pages :
0 1 2 3 4 5 6 7 8 9

 Enter the number of frame : 3
Ref string        page frames
0        0        -1        -1
1        0        1         -1
2        0        1         2
3        3        1         2
4        3        4         2
5        3        4         5
6        6        4         5
7        6        7         5
8        6        7         8
9        9        7         8

Page Fault Is 10

Process returned 0 (0x0)    execution time : 59.124 s
Press any key to continue.
```