Ex. No.: 11a)
Date: 12.04.25

FIFO PAGE REPLACEMENT

Aim:

To find out the number of page faults that occur using First-in First-out (FIFO) page replacement technique.

Algorithm:

1. Declare the size with respect to page length
2. Check the need of replacement from the page to memory
3. Check the need of replacement from old page to new page in memory 4.
Form a queue to hold all pages
5. Insert the page require memory into the queue
6. Check for bad replacement and page fault
7. Get the number of processes to be inserted
8. Display the values

Program Code:

```
#include<stdio.h>
int main()
{ int f, n, index=0, pt=0;
printf("Enter the size of reference string: ");
scanf("%d",&n);
int r[n];
for(int i=0;i<n;i++)
{ printf("Enter [%d]", i+1);
  scanf("%d",&r[i]);
}
printf("Enter Page frame size: ");
scanf("%d",&f);
int fr[f];
for(int i=0;i<f;i++){
  fr[i]=-1;
}
```

```
int found;
for(int i=0; i<f; i++)
{
found = 0;
printf("%d ⟶ ", r[i]);
for(int j=0; j<f; j++){
    if(fr[j] == r[i]) {
        found = 1;
        printf("No page fault");
        break;
    }
}
if(!found) {
    fr[index] = r[i];
    index = (index+1)%f;
    pf++;
    for(int k=0; k<f; k++) {
        if(fr[k] != -1)
            printf("%d ", fr[k]);
        else
            printf("- ");
    }
    printf("\n");
}
printf("Total page fault: %d", pf);
}
```

Enter the no of (size of reference string) : 12

Enter the page reference string

Page 1 : 7
Page 2 : 0
Page 3 : 1
Page 4 : 2
Page 5 : 0
Page 6 : 3
Page 7 : 0
Page 8 : 4
Page 9 : 2
Page 10 : 3
Page 11 : 0
Page 12 : 3

Enter the no of frames: 3

FIFO page replacement simulation

Page reference : 7 → Memory : 7
Page reference : 0 → Memory : 7 0
Page reference : 1 → Memory : 7 0 1
Page reference : 2 → Memory : 0 1 2
Page reference : 0 → Memory : 0 1 2
Page reference : 3 → Memory : 1 2 3
Page reference : 0 → Memory : 1 2 3
Page reference : 4 → Memory : 2 3 4
Page reference : 2 → Memory : 3 4 2
Page reference : 3 → Memory : 4 2 3
Page reference : 0 → Memory : 4 2 3
Page reference : 3 → Memory : 4 2 3

Total page faults : 9

Sample Output:

```
[root@localhost student]# python fifo.py

Enter the size of reference string: 20
Enter [ 1] : 7
Enter [ 2] : 0
Enter [ 3] : 1
Enter [ 4] : 2
Enter [ 5] : 0
Enter [ 6] : 3
Enter [ 7] : 0
Enter [ 8] : 4
Enter [ 9] : 2
Enter [10] : 3
Enter [11] : 0
Enter [12] : 3
Enter [13] : 2
Enter [14] : 1
Enter [15] : 2
Enter [16] : 0
Enter [17] : 1
Enter [18] : 7
Enter [19] : 0
Enter [20] : 1

Enter page frame size : 3

7 -> 7 - -
0 -> 7 0 -
1 -> 7 0 1
2 -> 2 0 1
0 -> No Page Fault


3 -> 2 3 1
0 -> 2 3 0
4 -> 4 3 0
2 -> 4 2 0
3 -> 4 2 3
0 -> 0 2 3
3 -> No Page Fault
2 -> No Page Fault
1 -> 0 1 3
2 -> 0 1 2
0 -> No Page Fault
1 -> No Page Fault
7 -> 7 1 2
0 -> 7 0 2
```

```
1 -> 7 0 1
Total page faults: 15.
[root@localhost student]#
```

Result :

Thus the C program to implement FIFO has been executed succesfully.