Ex. No.: 11a)
Date: 16/4/25

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 refstr[100], frames[10];
    int refsize, framesize;
    int index = 0, isHit, pf=0;
    printf("Enter size of ref string:");
    scanf("%d", &refsize);
    for(int i = 0; i <= refsize; i++){
        printf("Enter[%d]: ", i+1);
        scanf("%d", &refstr[i]);
    }
    printf("Enter page frame size:");
    scanf("%d", &framesize:");
    for(int i=0; i <refsize ; i++){
        isHit=0;
        for(int j=0; j< framesize; j++){
            if(frames[j] == refstr[i]){
                isHit=1;
                break;
            }
    }
```

```
if (!isHit){
    frames[index] = refstr[i];
    index = (index+1) % framesize;
    p++;
    printf("%d->", refstr[i]);
    for (int k=0; k<framesize; k++) {
        if (frames[k] != -1)
            print("%d", frames[k]);
    }
    printf("\n");
}
else{
    printf("%d -> No page faults\n", refstr[i]);
}
}
printf("\nTotal page faults: %d\n", pf);
}
```

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]#

OUTPUT:

Enter the size of ref string: 7

Enter page frame size: 3

Enter[1]: 1
Enter[2]: 3
Enter[3]: 0
Enter[4]: 3
Enter[5]: 5
Enter[6]: 1
Enter[7]: 3

1 → 1
3 → 13
0 → 130
3 → No page fault Total page faults: 6
7 → 530
6 → 560
3 → 563

Result :

A program for finding the page fault using FIFO replacement.