Ex. No.: 11 b
Date: 7.5.24

LRU

Aim:

To write a c program to implement LRU page replacement algorithm.

Algorithm:

1: Start the process
2: Declare the size
3: Get the number of pages to be inserted
4: Get the value
5: Declare counter and stack
6: Select the least recently used page by counter value
7: Stack them according the selection.
8: Display the values
9: Stop the process

Program Code: // lru.c

```
#include<stdio.h>
int findLRU(int time[], int n){
        int i, min = time[0], pos=0;
        for(i=0; i<n; ++i){
                if(time[i] < min){
                        min=time[i];
                        pos=i;
                }
        }
        return pos;
}
int main(){
    int no_of_frames, no_of_pages, counter=0, flag1, flag2, pos, i, j, faults=0;
    printf("Enter no of frames: ");
    scanf("%d", &no_of_frames);
    printf("Enter no of pages: ");
    scanf("%d",&no_of_pages);
    int frames[no_of_frames], pages[no_of_pages], time[no_of_frames];
    Printf("Enter reference string: ");
    for(i=0;i<no_of_pages;i++) scanf("%d", &pages[i]);
```

```
for(i=0;i<no_of_frames;i++){
        frames[i] = -1 ;
    }
for(i=0;i<no_of_pages ; ++i){
        flag1 = flag2 = 0 ;
        for(j=0 ;j<no_of_frames;++j) {
                if(frames[j] == pages[i]){
                        counter++;
                        time[j]=counter ;
                        flag1 = flag2 = 1;
                        break ;
                }
        }
        if(flag==0){
                for(j=0 ;j<no_of_frames ;++j) {
                        if(frames[j]==-1){
                                counter++;
                                faults++;
                                frames[j]= pages[i];
                                time[j] = counter ;
                                flag2=1 ;
                                break ;
                        }
                }
        }
if(flag2==0){
        pos = findLRU(time , no_of_frames);
        counter ++;
        faults++;
        frames[pos] = pages[i];
        time[pos] = counter ;
    }
printf("\n");
```

```
        for(j=0;j<no_of_frames;++j){
                printf("%d\t", frames[j]);
        }
    }
    printf("\n\nTotal Page Faults = %d\n", faults);
    return 0;
}
```

Output

```
gcc lru.c
./a.out

Enter no of frames: 3
Enter no of pages: 6
Enter reference string: 5 7 5 6 7 3

5       -1      -1
5       7       -1
5       7       -1
5       7       6
5       7       6
3       7       6

Total page faults = 4
```

RESULT:

The program was been compiled and executed successfully.

7/5/24