Ex. No.: 11b)
Date: 17/04/25

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:

```
#include <stdio.h>
int findLRU (int time[], int n) {
        int i, min = time[0], pos=0;
        for (i=1; i<n; ++i) {
            if (time[i] < min) {
                min = time[i];
                pos = i;
            }
        }
        return pos;
}
int main () {
    int frames, pages, i, j, counter=0, flag1, flag2, page_faults=0;
    printf ("Enter the no. of frames: ");
    scanf ("%d", &frames);
    printf ("Enter the no. of pages: ");
    scanf ("%d", &pages);
    int incoming[pages], temp[frames], time[frames];
    printf ("Enter page reference string");
```

```
for(i = 0; i < Pages; i++)
        scanf("%d", &incoming[i]);
for (i=0; i< frames; i++){
        temp[i] = -1;
        time[i] = 0;
}
printf("\nPage\t Frame1 \t Frame2\t Frame3\t Page faults\n");
for(i=0; i< pages; i++){
    flag1 = flag2 =0;
    for(j=0; j< frames; j++){
        if(temp[j] == incoming[i]){
            counter++;
            time[j] = counter;
            flag1 = flag2 = 1;
            break;
        }
    }
    if(flag1 == 0){
        for(j=0; j< frames; j++){
            if(temp[j] == -1){
                counter++;
                page-faults++;
                temp[j] = incoming[i];
                time[j] = counter;
                flag2 = 1;
                break;
            }
        }
    }
```

```
if (flag2 == 0) {
        int pos = findLRU(time, frames);
        counter++;
        page_faults++;
        temp[pos] = incoming[i];
        time[pos] = counter;
}
Printf("%d\t", incoming[i]);
for(j=0; j<frames; j++) {
        if(temp[j] != -1)
                printf("%d\t", temp[j]);
        else
                printf("-1\t");
}
if(flag1 == 0){
        printf("1\n");
} else {
        printf("0\n");
}
}
printf("\nTotal Page Faults : %d\n", page_faults);
return 0;
}
```

Sample Output :

Enter number of frames: 3
Enter number 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

Enter no. of frames = 3
Enter no. of pages = 6
Enter page reference string = 5 7 5 6 7 3

Page	Frame1	Frame2	Frame 3	Page Faults
5	5	-	-	1
7	5	7	-	1
5	5	7	-	0
6	5	7	6	1
7	5	7	6	0
3	5	7	6	1

Total Page Faults : 4

Result:

Thus the program to find out the number of Page faults that occur using Least Recently Used (LRU) page replacement technique has been executed successfully.