Ex. No.: 11b LRU Page Replacement Date:

15.04.2025 roll no:231901062

Aim:

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

Algorithm:

1. Start the process.
2. Declare the size for page frames.
3. Get the number of pages and reference string.
4. Use a stack or counter array to track recent usage.
5. For each page:
    - If it is in memory → no page fault.
    - Else → check least recently used page and replace it.
6. Count page faults.
7. Display frame contents after each operation.
8. Stop the process.

C Program:

```
#include <stdio.h>

int findLRU(int time[], int n) {
 int i, minimum = time[0], pos = 0;
 for(i = 1; i < n; i++) {
 if(time[i] < minimum) {
 minimum = time[i];
 pos = i;
 }
 }
```

```
 return pos;
}

int main() {
 int frames[10], pages[50], time[10], counter = 0, pageFaults = 0;
int n, f, i, j, pos, flag1, flag2;

 printf("Enter number of frames: ");
 scanf("%d", &f);

 printf("Enter number of pages: ");
 scanf("%d", &n);

 printf("Enter reference string: ");
 for(i = 0; i < n; i++)
 scanf("%d", &pages[i]);

 for(i = 0; i < f; i++)
 frames[i] = -1;

 for(i = 0; i < n; i++) {
 flag1 = flag2 = 0;

 for(j = 0; j < f; j++) {
 if(frames[j] == pages[i]) {
 counter++;
 time[j] = counter;
 flag1 = flag2 = 1;
```

```
 break;
 }
 }

 if(flag1 == 0) {
 for(j = 0; j < f; j++) {
if(frames[j] == -1) {
counter++;
 pageFaults++;  frames[j] =
pages[i];  time[j] = counter;
flag2 = 1;
 break;
 }
 }
 }

 if(flag2 == 0) {
 pos = findLRU(time, f);
counter++;
 pageFaults++;
 frames[pos] = pages[i];
time[pos] = counter;  }

 for(j = 0; j < f; j++) {
 if(frames[j] != -1)
 printf("%d ", frames[j]);  else
 printf("- ");
 }
```

```
 printf("\n");
 }

 printf("\nTotal Page Faults = %d\n", pageFaults);
 return 0;
}
```

Sample Output:

Enter number of frames: 3

Enter number of pages: 6

Enter reference string: 5 7 5 6 7 3

5 - -

5 7 -

5 7 -

5 7 6

5 7 6

3 7 6

Total Page Faults = 4

Result:

Thus, the C program for LRU page replacement algorithm was written and executed successfully. The number of page faults was calculated and verified.