Ex. No.: 11b)
Date: 09-04-2025

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

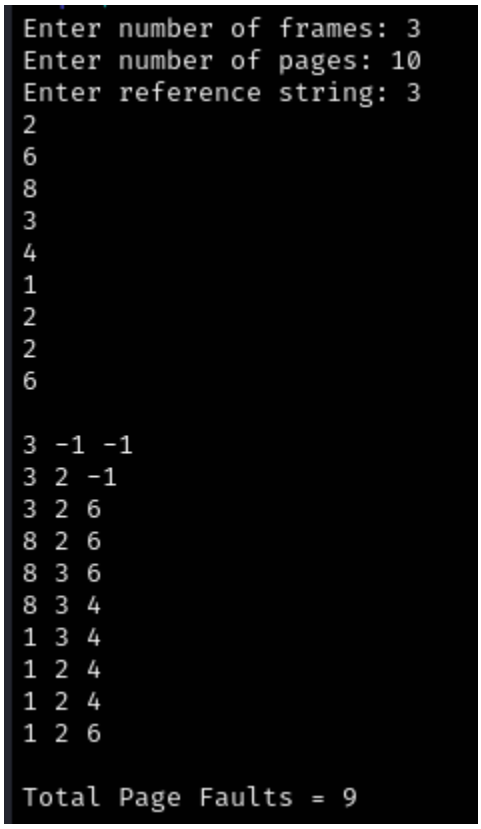
Program Code:

```
#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[10], pages[30], counter[10];
        int i, j, k, pos, max, faults = 0, time = 0;
        int n, f;
        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;
counter[i] = 0;
}
printf("\n");
for (i = 0; i < n; ++i) {
int flag1 = 0, flag2 = 0;
for (j = 0; j < f; ++j) {
if (frames[j] == pages[i]) {
        time++;
        counter[j] = time; // Update recent use time
        flag1 = flag2 = 1;
        break;
}
}
if (flag1 == 0) {
for (j = 0; j < f; ++j) {
        if (frames[j] == -1) {
        time++;
        faults++;
        frames[j] = pages[i];
        counter[j] = time;
        flag2 = 1;
        break;
        }
}
}
if (flag2 == 0) {
pos = findLRU(counter, f);
time++;
faults++;
frames[pos] = pages[i];
counter[pos] = time;
}
// Display current frame state
for (k = 0; k < f; ++k) {
if (frames[k] != -1)
        printf("%d ", frames[k]);
else
        printf("-1 ");
}
```

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

OUTPUT:



RESULT:

Hence, page faults that occur using LRU page replacement technique has been found.