

OPERATING SYSTEM - CS23431

EXP 11(B)

LRU PAGE REPLACEMENT

NAME: S.KUMARAN

ROLL NO: 230701159

PROGRAM:

```
#include <stdio.h>
```

```
int main() {  
    int n, frame_size, count = 0, page_faults = 0;
```

```
    printf("Enter size of reference string: ");  
    scanf("%d", &n);
```

```
    int page[n];  
    for (int i = 0; i < n; i++) {  
        printf("Enter [%d]: ", i + 1);  
        scanf("%d", &page[i]);  
    }
```

```
    printf("Enter page frame size: ");  
    scanf("%d", &frame_size);
```

```
    int mem[frame_size];
```

```
    for (int i = 0; i < n; i++) {  
        int top = -1;  
        int f = 0;
```

```
        // Check if page already in memory
```

```
        for (int j = 0; j < count; j++) {  
            if (mem[j] == page[i]) {  
                top = j;  
                break;  
            }  
        }
```

```
        printf("%d -> ", page[i]);
```

```

if (top != -1) {
    // If found, move it to most recently used position (end)
    int temp = mem[top];
    for (int j = top; j < count - 1; j++) {
        mem[j] = mem[j + 1];
    }
    mem[count - 1] = temp;
    printf("No page fault\n");
} else {
    f = 1;
    if (count < frame_size) {
        mem[count++] = page[i];
    } else {
        // Remove least recently used (front), shift, add new at end
        for (int j = 0; j < frame_size - 1; j++) {
            mem[j] = mem[j + 1];
        }
        mem[frame_size - 1] = page[i];
    }
    page_faults++;
}

if (f) {
    for (int j = 0; j < count; j++) {
        printf("%d ", mem[j]);
    }
}

printf("\n");
}

printf("\nTotal Page Faults: %d\n", page_faults);
return 0;
}

```

OUTPUT:

```
Enter size of reference string: 6
Enter [1]: 5
Enter [2]: 7
Enter [3]: 5
Enter [4]: 6
Enter [5]: 7
Enter [6]: 3
Enter page frame size: 3
5 -> 5
7 -> 5 7
5 -> No page fault

6 -> 7 5 6
7 -> No page fault

3 -> 6 7 3

Total Page Faults: 4
```