LRU

```
#include <stdio.h>

void display(int frames[], int n) {
    for (int i = 0; i < n; i++) {
        printf(frames[i] == -1 ? " - " : " %d ", frames[i]);
    }
    printf("\n");
}

void displayFaultFrames(int fault_frames[], int fault_count) {
    printf("Fault Frames: ");
    for (int i = 0; i < fault_count; i++) {
        printf("%d ", fault_frames[i]);
    }
    printf("\n");
}

int isPageInFrame(int page, int frames[], int n) {
    for (int i = 0; i < n; i++) {
        if (frames[i] == page) return i;
    }
    return -1;
}

int findLRU(int time[], int n) {
    int min_time = time[0], index = 0;
    for (int i = 1; i < n; i++) {
        if (time[i] < min_time) {
            min_time = time[i];
            index = i;
        }
    }
    return index;
}

void runLRU(int ref_string[], int ref_len, int n) {
    int frames[n], page_faults = 0, time[n];
    int fault_frames[n]; // To store pages causing faults
    int fault_count = 0;

    for (int i = 0; i < n; i++) {
        frames[i] = -1;
        time[i] = 0;
    }

    printf("Page Replacement Process (LRU):\n");
    for (int i = 0; i < ref_len; i++) {
        int page = ref_string[i];
        int index = isPageInFrame(page, frames, n);
        if (index == -1) {
            int lru_index = findLRU(time, n);
            frames[lru_index] = page;
            fault_frames[fault_count++] = page; // Store fault page
            page_faults++;
        }
        for (int j = 0; j < n; j++) {
            if (frames[j] == page) time[j] = i + 1;
        }
        display(frames, n);
    }
    printf("Total Page Faults: %d\n", page_faults);
    displayFaultFrames(fault_frames, fault_count); // Display fault frames
}

int main() {
    int n;
    printf("Enter number of frames: ");
    scanf("%d", &n);

    int ref_string[] = {3, 5, 7, 2, 5, 1, 2, 3, 1, 3, 5, 3, 1, 6, 2};
    int ref_len = sizeof(ref_string) / sizeof(ref_string[0]);

    runLRU(ref_string, ref_len, n);
    return 0;
}
```