Experiment-32:Construct a C program to simulate the Least Recently Used paging technique of memory management

Aim:
To simulate the Least Recently Used (LRU) paging technique of memory management in C.

Procedure:

1. Take the number of pages and the number of frames as input.
2. Simulate the LRU algorithm by tracking the order of page accesses.
3. If a page is not in memory, replace the least recently used page with the new one.
4. Keep track of page faults and display the results.

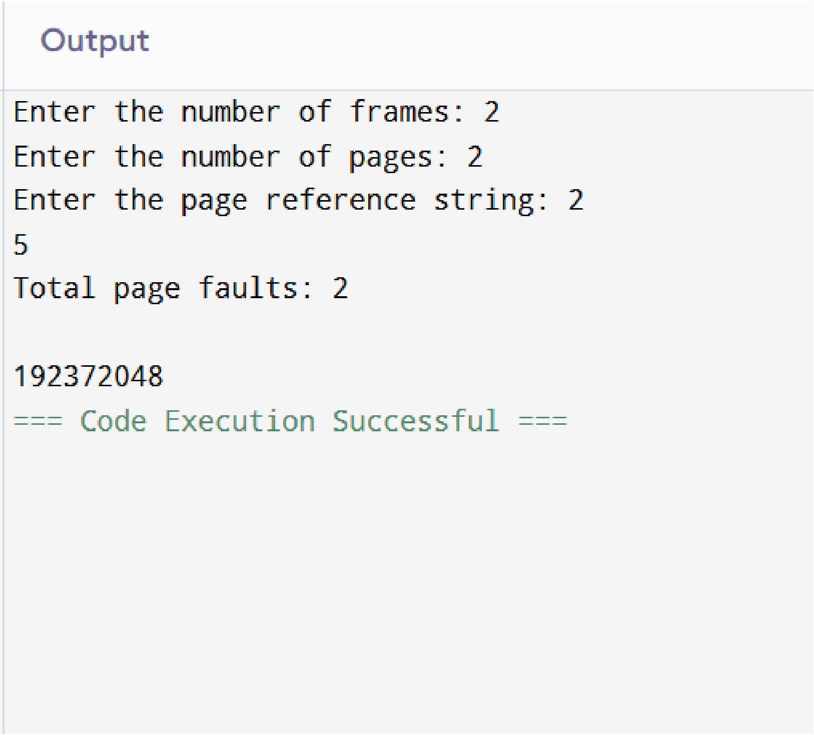
C Program:

```
#include <stdio.h>


int main() {
    int frames, pages, page_faults = 0;
    printf("Enter the number of frames: ");
    scanf("%d", &frames);
    printf("Enter the number of pages: ");
    scanf("%d", &pages);


    int page_sequence[pages], frame[frames], time[frames];
    for (int i = 0; i < frames; i++) {
        frame[i] = -1;
        time[i] = -1;
    }


    printf("Enter the page reference string: ");
    for (int i = 0; i < pages; i++) {
        scanf("%d", &page_sequence[i]);
    }


    for (int i = 0; i < pages; i++) {
        int page_found = 0, min_time = 0, replace_index = -1;
```

```
for (int j = 0; j < frames; j++) {
    if (frame[j] == page_sequence[i]) {
        page_found = 1;
        time[j] = i;
        break;
    }
}

if (!page_found) {
    for (int j = 0; j < frames; j++) {
        if (frame[j] == -1) {
            frame[j] = page_sequence[i];
            time[j] = i;
            page_faults++;
            break;
        }
    }
    if (page_faults <= frames) continue;

    for (int j = 0; j < frames; j++) {
        if (time[j] < time[min_time]) {
            min_time = j;
            replace_index = j;
        }
    }
    frame[replace_index] = page_sequence[i];
    time[replace_index] = i;
    page_faults++;
}

printf("Frame state after page %d: ", page_sequence[i]);
```

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

    printf("Total page faults: %d\n", page_faults);
    return 0;
}
```

Output:



```
Enter the number of frames: 2
Enter the number of pages: 2
Enter the page reference string: 2
5
Total page faults: 2

192372048
=== Code Execution Successful ===
```