Experiment-31:Construct a C program to simulate the First in First Out paging technique of memory management.

Aim:
To simulate the First In First Out (FIFO) paging technique of memory management in C.

Procedure:

1. Take the number of pages and the number of frames as input.
2. Simulate the FIFO algorithm by storing pages in frames.
3. If a page needs to be loaded and all frames are occupied, replace the page that has been in memory the longest.
4. Keep track of page faults and display the results.

C Program:

```
#include <stdio.h>


int main() {
   int frames, pages, page_faults = 0, pointer = 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];
   for (int i = 0; i < frames; i++) {
      frame[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;
    for (int j = 0; j < frames; j++) {
        if (frame[j] == page_sequence[i]) {
            page_found = 1;
            break;
        }
    }

    if (!page_found) {
        frame[pointer] = page_sequence[i];
        pointer = (pointer + 1) % frames;
        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:

Output

```
Enter the number of frames: 2
Enter the number of pages: 2
Enter the page reference string: 1
2
Frame state after page 1: 1  -
Frame state after page 2: 1 2
Total page faults: 2
```