2.Write a C program to illustrate the FIFO method of page replacement and determine the number of page faults for the following test case:

No of page frames: 3; Page reference sequence: 4, 1, 2, 4, 3, 2, 1 and 5.

PROGRAM:

#include <stdio.h>

#define MAX\_FRAMES 10

int main() {

int pages[MAX\_FRAMES]; // array to store the pages in frames

int page\_count = 0; // number of pages currently in frames

int page\_faults = 0; // number of page faults

int i, j, k;

// initialize all pages to -1 to indicate an empty frame

for (i = 0; i < MAX\_FRAMES; i++) {

pages[i] = -1;

}

// read the page sequence from input

int n;

printf("Enter number of pages: ");

scanf("%d", &n);

int sequence[n];

printf("Enter page sequence: ");

for (i = 0; i < n; i++) {

scanf("%d", &sequence[i]);

}

// perform FIFO page replacement

for (i = 0; i < n; i++) {

int page = sequence[i];

int page\_found = 0;

// check if page is already in frames

for (j = 0; j < page\_count; j++) {

if (pages[j] == page) {

page\_found = 1;

break;

}

}

// if page is not in frames, replace the oldest page with the new page

if (!page\_found) {

if (page\_count < MAX\_FRAMES) {

// add new page to an empty frame

pages[page\_count] = page;

page\_count++;

} else {

// replace the oldest page with the new page

for (j = 0; j < MAX\_FRAMES - 1; j++) {

pages[j] = pages[j+1];

}

pages[MAX\_FRAMES - 1] = page;

}

page\_faults++;

}

// print the current state of the frames after each page is processed

printf("Page %d: ", page);

for (j = 0; j < page\_count; j++) {

printf("%d ", pages[j]);

}

for (j = page\_count; j < MAX\_FRAMES; j++) {

printf("- ");

}

printf("\n");

}

printf("Total number of page faults: %d\n", page\_faults);

return 0;

}

OUTPUT:

