1. Construct a C program to simulate the Least Recently Used paging technique of memory management.

#include <stdio.h>

int findLRU(int time[], int n) {

int i, min = time[0], pos = 0;

for (i = 1; i < n; i++) {

if (time[i] < min) {

min = time[i];

pos = i;

}

}

return pos;

}

int main() {

int pages[30], frames[10], time[10];

int n, f, i, j, pos, faults = 0, counter = 0;

int found;

printf("Enter number of pages: ");

scanf("%d", &n);

printf("Enter the page reference string:\n");

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

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

printf("Enter number of frames: ");

scanf("%d", &f);

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

frames[i] = -1;

time[i] = 0;

}

printf("\nPage\tFrames\t\tPage Fault\n");

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

found = 0;

for (j = 0; j < f; j++) {

if (frames[j] == pages[i]) {

counter++;

time[j] = counter;

found = 1;

break;

}

}

if (!found) {

for (j = 0; j < f; j++) {

if (frames[j] == -1) {

counter++;

frames[j] = pages[i];

time[j] = counter;

faults++;

found = 1;

break;

}

}

}

if (!found) {

pos = findLRU(time, f);

counter++;

frames[pos] = pages[i];

time[pos] = counter;

faults++;

}

printf("%d\t", pages[i]);

for (j = 0; j < f; j++) {

if (frames[j] != -1)

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

else

printf("- ");

}

printf("\t\t%s\n", found ? "No" : "Yes");

}

printf("\nTotal Page Faults = %d\n", faults);

return 0;

}