**Name:- Sachin Gurjar**

**Section:- A**

**Enrollment No:- A20405220165**

**Practical Exam**

Experiment-2

Aim:- Write a C Program to implementation of LRU page replacement policy.

**Software used:** Code-blocks

**Code:-**

#include<stdio.h>

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

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

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

if(time[i] < minimum){

minimum = time[i];

pos = i;

}

}

return pos;

}

int main()

{

int no\_of\_frames, no\_of\_pages, frames[10], pages[30], counter = 0, time[10], flag1, flag2, i, j, pos, faults = 0;

printf("Enter number of frames: ");

scanf("%d", &no\_of\_frames);

printf("Enter number of pages: ");

scanf("%d", &no\_of\_pages);

printf("Enter reference string: ");

for(i = 0; i < no\_of\_pages; ++i){

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

}

for(i = 0; i < no\_of\_frames; ++i){

frames[i] = -1;

}

for(i = 0; i < no\_of\_pages; ++i){

flag1 = flag2 = 0;

for(j = 0; j < no\_of\_frames; ++j){

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

counter++;

time[j] = counter;

flag1 = flag2 = 1;

break;

}

}

if(flag1 == 0){

for(j = 0; j < no\_of\_frames; ++j){

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

counter++;

faults++;

frames[j] = pages[i];

time[j] = counter;

flag2 = 1;

break;

}

}

}

if(flag2 == 0){

pos = findLRU(time, no\_of\_frames);

counter++;

faults++;

frames[pos] = pages[i];

time[pos] = counter;

}

printf("\n");

for(j = 0; j < no\_of\_frames; ++j){

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

}

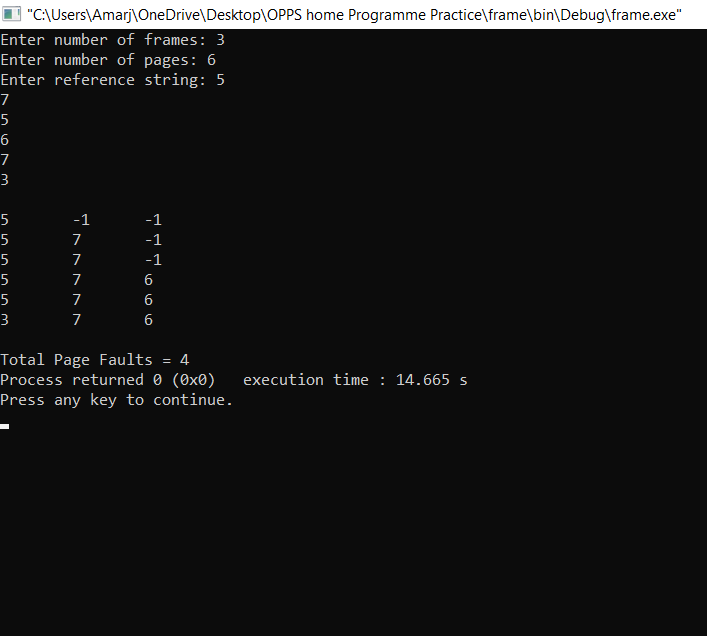
}

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

return 0;

}

**OUTPUT**

****

Experiment-1

Aim:-Write a C program to count the occurrence of each word and total words in a file.

**Software used:** Visual Studio Code

**Code:-**

#include <stdio.h>

#define MAX\_LEN 1024

int main() {

char ch;

int char\_count = 0, word\_count = 0, line\_count = 0;

int in\_word = 0;

char file\_name[MAX\_LEN];

FILE \*fp;

printf("Enter a file name: ");

scanf("%s", file\_name);

fp = fopen(file\_name, "r");

if(fp == NULL) {

printf("Could not open the file %s\n", file\_name);

return 1;

}

while ((ch = fgetc(fp)) != EOF) {

char\_count++;

if(ch == ' ' || ch == '\t' || ch == '\0' || ch == '\n') {

if (in\_word) {

in\_word = 0;

word\_count++;

}

if(ch = '\0' || ch == '\n') line\_count++;

} else {

in\_word = 1;

}

}

printf("In the file %s:\n", file\_name);

printf("Number of characters: %d.\n", char\_count);

printf("Number of words: %d.\n", word\_count);

printf("Number of lines: %d.\n", line\_count);

return 0;

}

**OUTPUT**

Enter file path: sachin.txt

Number of words: 9.

Occurrences of all distinct words in file:

my => 1

name => 1

is => 1

sachin => 1

gurjar => 1

i => 1

live => 1

in => 1

delhi => 1