**EXPERIMENT- 32**

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

management.

#include <stdio.h>

#define MAX\_FRAMES 10

#define MAX\_REFERENCES 100

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

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

for (int i = 1; i < frameCount; i++) {

if (time[i] < min) {

min = time[i];

pos = i;

}

}

return pos;

}

void simulateLRU(int pages[], int n, int frameCount) {

int frames[MAX\_FRAMES], time[MAX\_FRAMES];

int pageFaults = 0, count = 0;

for (int i = 0; i < frameCount; i++) {

frames[i] = -1;

time[i] = 0;

}

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

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

int page = pages[i], flag = 0;

for (int j = 0; j < frameCount; j++) {

if (frames[j] == page) {

count++;

time[j] = count;

flag = 1;

break;

}

}

if (!flag) {

int pos = -1;

for (int j = 0; j < frameCount; j++) {

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

pos = j;

break;

}

}

if (pos == -1)

pos = findLRU(time, frameCount);

frames[pos] = page;

count++;

time[pos] = count;

pageFaults++;

}

printf("%d\t", page);

for (int j = 0; j < frameCount; j++) {

if (frames[j] == -1)

printf("- ");

else

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

}

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

}

printf("\nTotal Page Faults: %d\n", pageFaults);

}

int main() {

int pages[MAX\_REFERENCES], n, frameCount;

printf("Enter number of pages: ");

scanf("%d", &n);

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

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

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

printf("Enter number of frames: ");

scanf("%d", &frameCount);

simulateLRU(pages, n, frameCount);

return 0;

}

SAMPLE INPUT:

Enter number of pages: 12

Enter the page reference string:

1 2 3 4 1 2 5 1 2 3 4 5

Enter number of frames: 3

SAMPLE OUTPUT:

Page Frames Page Fault

1 1 - - Yes

2 1 2 - Yes

3 1 2 3 Yes

4 4 2 3 Yes

1 4 1 3 Yes

2 4 1 2 Yes

5 5 1 2 Yes

1 5 1 2 No

2 5 1 2 No

3 3 1 2 Yes

4 3 4 2 Yes

5 3 4 5 Yes

Total Page Faults: 10