****

**Green University of Bangladesh**

**Department of Computer Science and Engineering (CSE)**

**Semester: (Spring, Year:2024), B.Sc. in CSE (Day)**

**Lab Report NO #05**

**Course Title:** Operating System Lab

**Course Code:** CSE 310  **Section:** 221 D6

**Lab Experiment Name:** Implementation of LRU page replacement algorithm.

**Student Details**

| **Name** | | **ID** |
| --- | --- | --- |
| **1.** | Tanvir Ahmed | 221002461 |

**Lab Date : 13/5/2024**

**Submission Date : 20/5/2024**

**Course Teacher’s Name : Jarin Tasnim Tonvi**

| **Lab Report Status**  **Marks: ………………………………… Signature:.....................**  **Comments:.............................................. Date:..............................** |
| --- |

**1. TITLE OF THE LAB REPORT EXPERIMENT**

Implementation of Least Recently Used (LRU) page replacement algorithm.

**2. OBJECTIVES/AIM**

* Understand the Least Recently Used page replacement algorithm.
* To implement the Least Recently Used page replacement algorithm using c.

**3. PROCEDURE**

* First, take the number of pages, pages, and number of frames from the user.
* Run a loop for the frames, and first check whether the frame hit or not.
* If hit, just print the value in the frame.
* If not, check whether the frame is empty or not. If empty, assign the page to the frame and print the frame values.
* If the frame is not empty, backtrack the pages to find the least recently used page.
* After finding the least recently used page, replace the corresponding frame with the new page and print the frame values.
* At last print the page fault that I count at the beginning of the code. Print the page hit also using the page fault.

**4. IMPLEMENTATION**

Source Code:

#**include**<stdio.h>

int **main**(){

int p, f;

printf("Enter the number of pages: ");

scanf("%d", &p);

int pages[p];

printf("Enter the pages: ");

**for**(int i = 0; i < p; i++){

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

}

printf("Enter the number of frames: ");

scanf("%d", &f);

int frames[f];

int backtrack[f];

int temp = 0;

int pagefault = 0;

printf("Page\t\t Frame1 \t \tFrame2 \t\t Frame3\n");

**for**(int i = 0; i < p; i++){

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

int hit = 0;

**for**(int j= 0; j < temp; j++){

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

hit = 1;

**break**;

}

}

**if**(hit==1)

**for**(int i = 0; i < temp; i++)

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

**else** **if**(temp < f){

frames[temp] = pages[i];

pagefault++;

temp++;

**for**(int i = 0; i < temp; i++)

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

}

**else** {

int max = 0;

int md;

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

backtrack[j] = 0;

**for**(int k = i - 1; k >= 0; k--){

**if**(frames[j] == pages[k]){

backtrack[j]++;

**break**;

}

backtrack[j]++;

}

**if**(backtrack[j] > max){

max = backtrack[j];

md = j;

}

}

frames[md] = pages[i];

**for**(int i = 0; i < temp; i++)

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

pagefault++;

}

printf("\n");

}

printf("Page Fault: %d\n", pagefault);

printf("Page Hit: %d", p-pagefault);

**return** 0;

}

**5. TEST RESULT / OUTPUT**

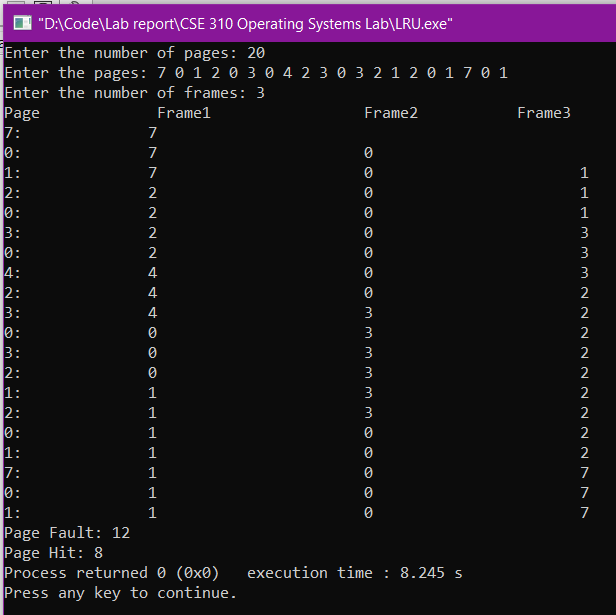
****

Fig1. Sample input and output.

**6. ANALYSIS AND DISCUSSION**

* In this exercise, I implemented the least recently used page replacement algorithm.
* In the previous lab class, I learned and implemented the FIFO page replacement algorithm and then I explained the LRU algorithm from the internet source.
* I implement the same method here that I explained in the previous class.
* While implementing I got ideas from the lab report (FIFO algorithm) and from the internet which I explained in my previous class.
* Finally, I succeeded in implementing the LRU algorithm in this exercise.

**7. SUMMARY**

This lab exercise demonstrates the successful implementation of the least recently used page replacement algorithm based on the knowledge I learned from the previous lab class. In the last class, I learned and implemented the first in first out page replacement algorithm and learned LRU and LFU basics. Then I explained a method to implement the LRU page replacement algorithm. I implement the same method here for the LRU page replacement algorithm. I implemented the algorithm without any major difficulties, I faced some syntax errors and I was able to solve this instantly.