

Smt. Chandibai Himathmal Mansukhani College

Contents

USCS3P01:USCS303-Operating System (OS) Practical-09	2
Practical – 09: Page Replacement Algorithm Least Recently Used (LRU)	2
Practical Date: 31-08-2021	2
Practical Aim: Page Replacement Algorithm (LRU)	2
Algorithm:	2
Page Replacement Algorithm	2
Least Recently Used (LRU) Algorithm	2
Solved Example:	2
Example-01:	2
Example-02:	4
Example-03:	5
Question:	6
Implementation:	6
Input 01:	Error! Bookmark not defined.
Output 01:	10
Sample Output 01:	Error! Bookmark not defined.
Input 02:	10
Output 02:	11
Sample Output 02:	Error! Bookmark not defined.
Input 03:	11
Output 03:	11
Sample Output 03:	Error! Bookmark not defined.

USCS3P01:USCS303-Operating System (OS) Practical-09

Practical – 09: Page Replacement Algorithm Least Recently Used (LRU)

Practical Date: 31-08-2021

Practical Aim: Page Replacement Algorithm (LRU)

Algorithm:

Page Replacement Algorithm

In demand paging memory management technique, if a page demanded for execution is present in main memory, then a page fault occurs.

To load the page in demand into main memory, a free page frame is searched in main memory and allocated.

If no page frame is free, memory manager has to free a frame by swapping its contents to secondary storage and thus make room for the required page.

To swap pages, many schemes or strategies are used.

Least Recently Used (LRU) Algorithm

The **Least Recently Used (LRU) algorithm** replaces the page that has not been used for the longest period of time.

It is based on the observation that pages that have not been used for long time will probably remain unused for the longest time and are to be replaced.

Solved Example:

Example-01:

Apply the LRU replacement algorithms for the following page-reference strings:

7,0,1,2,0,3,0,4,2,3,0,3,2

Indicate the number of page faults for LRU algorithm assuming demand paging with four frames.

Find the number of hits, number of faults and hit ratio

Solution

Page Reference String: 7,0,1,2,0,3,0,4,2,3,0,3,2

Demand Paging or Number of Frames: 4

Smt. Chandibai Himathmal Mansukhani College

7	7	7	7	7	3	3	3	3	3	3	3
-1	0	0	0	0	0	0	0	0	0	0	0
-1	-1	1	1	1	1	4	4	4	4	4	4
-1	-1	-1	2	2	2	2	2	2	2	2	2

7	0	1		2	0	3	0	4	2	3	0	3	2
×	×	×		×	✓	×	✓	×	✓	✓	✓	✓	✓

Number of Hits: Count of no replacements = 7 ✓

Number of Faults: Count of replacements = 6 ✗

Hit Ratio: Number of Hits/Len (Ref String) = $7/13 = 0.53$

Smt. Chandibai Himathmal Mansukhani College

Example-02:

Consider the following example 3 frames with 1,3,0,3,5,6,3 page-reference strings.

Find the number of hits, number of faults and hit ratio using LRU Page Replacement Algorithm.

Solution:

Page Reference String: 1,3,0,3,5,6,3

Demand Paging or Number of Frames: 3

1	1	1	1	5	5	5
-1	3	3	3	3	3	3
-1	-1	0	0	0	6	6

1	3	0	3	5	6	3
×	×	×	×	×	×	✓

Number of Hits: Count of no replacements = 2 ✓

Number of Faults: Count of replacements = 5 ✗

Hit Ratio: Number of Hits/Len (Ref String) = $2/7 = 0.28$

Smt. Chandibai Himathmal Mansukhani College

Example-03:

Consider the following example 3 frames with 7,0,1,2,0,3,0,4,2,3,0,3,2,1,2,0,1,7,0,1 page-reference strings.

Find the number of hits, number of faults and hit ratio using LRU Page Replacement Algorithm.

Solution:

Page Reference String: 7,0,1,2,0,3,0,4,2,3,0,3,2,1,2,0,1,7,0,1

Demand Paging or Number of Frames: 3

7	7	7	2	2	2	2	4	4	4	0	0	0	1	1	1	1	1	1	1
-1	0	0	0	0	0	0	0	0	3	3	3	3	3	3	0	0	0	0	0
-1	-1	1	1	1	3	3	3	2	2	2	2	2	2	2	2	2	7	7	7

7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
×	×	×	×	✓	×	✓	×	×	×	×	✓	✓	×	✓	×	✓	×	✓	✓

Number of Hits: Count of no replacements = 8 ✓

Number of Faults: Count of replacements = 12 ✗

Hit Ratio: Number of Hits/Len (Ref String) = $8/20 = 0.4$

Smt. Chandibai Himathmal Mansukhani College

Question:

Write a Java program that implements the LRU page-replacement algorithm.

Implementation:

```
//Name: ABHISHEKNIKAM
// Batch: B2
// PRN: 2020016400805951
// Date: 31 August,2021
// Prac-09: Page Replacement Algorithm(LRU)

import java.io.*;
import java.util.*;

public class P9_PR_LRU_AN
{
    public static void main(String[] args) throws IOException
    {
        Scanner scan = new Scanner(System.in);

        int frames,pointer = 0, hit = 0, fault = 0,ref_len;

        Boolean isFull = false;

        int buffer[];

        ArrayList<Integer>stack = new ArrayList<Integer>();

        int reference[];

        int mem_layout[][];

        System.out.print("Please enter the number of Frames: ");

        frames = scan.nextInt();

        System.out.print("Please enter the length of the References strings: ");

        ref_len = scan.nextInt();
```

Smt. Chandibai Himathmal Mansukhani College

```
reference = new int[ref_len];
mem_layout = new int[ref_len][frames];
buffer = new int[frames];
for(int j = 0; j < frames; j++)
    buffer[j] = -1;
System.out.print("Please enter the references strings: ");
for(int i = 0; i < ref_len; i++)
{
    reference[i] = scan.nextInt();
}
System.out.println();
for(int i = 0; i < ref_len; i++)
{
    if(stack.contains(reference[i]))
    {
        stack.remove(stack.indexOf(reference[i]));
    }
    stack.add(reference[i]);
    int search = -1;
    for(int j = 0; j < frames; j++)
    {
        if(buffer[j] == reference[i])
        {
            search = j;
            hit++;
            break;
        }
    }
    if(search == -1)
    {
```

```
        if(isFull)
        {
            int min_loc = ref_len;
            for(int j = 0; j < frames; j++)
            {
                if(stack.contains(buffer[j]))
                {
                    int temp = stack.indexOf(buffer[j]);
                    if(temp < min_loc)
                    {
                        min_loc = temp;
                        pointer = j;
                    }
                }
            }
            buffer[pointer] = reference[i];
            fault++;
            pointer++;
            if(pointer == frames)
            {
                pointer = 0;
                isFull = true;
            }
        }
        for(int j = 0; j < frames; j++)
            mem_layout[i][j] = buffer[j];
    }
    for(int i = 0; i < frames; i++)
    {
```


Smt. Chandibai Himathmal Mansukhani College

```
        for(int j = 0; j < ref_len; j++)  
            System.out.printf("%3d",mem_layout[j][i]);  
        System.out.println();  
    }  
  
    System.out.println("The number of Hits: " + hit);  
    System.out.println("Hit Ratio: " +(float)((float)hit/ref_len));  
    System.out.println("The number of Faults: " + fault);  
}  
}
```

Smt. Chandibai Himathmal Mansukhani College

Question 1 :

Output 01:

```
Please enter the number of Frames: 4
Please enter the length of the References strings: 13
Please enter the references strings:
7 0 1 2 0 3 0 4 2 3 0 3 2

  7 7 7 7 7 3 3 3 3 3 3 3 3
-1 0 0 0 0 0 0 0 0 0 0 0 0
-1 -1 1 1 1 1 1 4 4 4 4 4 4
-1 -1 -1 2 2 2 2 2 2 2 2 2 2
The number of Hits: 7
Hit Ratio: 0.53846157
The number of Faults: 6
```

Question 02:

Smt. Chandibai Himathmal Mansukhani College

Output 02:

```
Please enter the number of Frames: 3
Please enter the length of the References strings: 7
Please enter the references strings:
1 3 0 3 5 6 3

  1  1  1  1  5  5  5
-1  3  3  3  3  3  3
-1 -1  0  0  0  6  6
The number of Hits: 2
Hit Ratio: 0.2857143
The number of Faults: 5
```

Question 03:

Output 03:

```
Please enter the number of Frames: 3
Please enter the length of the References strings: 20
Please enter the references strings:
7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 1
0

  7  7  7  2  2  2  2  4  4  4  0  0  0  1  1  1  1  1  1  1
-1  0  0  0  0  0  0  0  0  0  3  3  3  3  3  3  0  0  0  0
-1 -1  1  1  1  3  3  3  2  2  2  2  2  2  2  2  7  7  7  7
The number of Hits: 8
Hit Ratio: 0.4
The number of Faults: 12
```