**Batch: B3 Roll No.: 121**

**Experiment / assignment / tutorial No.\_\_\_6\_\_\_**

**Grade: AA / AB / BB / BC / CC / CD /DD**

**Signature of the Staff In-charge with date**

|  |
| --- |
| **TITLE :**Implementation ofFIFO Page Replacement Algorithm |

**AIM:** The FIFO algorithm uses the principle that the block in the set which has been in for the longest time will be replaced

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Expected OUTCOME of Experiment: (Mention CO/CO’s attained here)**

CO1 – Describe and define the structure of a computer with buses structure and detail working of

the arithmetic logic unit and its sub modules.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Books/ Journals/ Websites referred:**

1. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, “Computer Organization”, Fifth Edition, TataMcGraw-Hill.
2. William Stallings, “Computer Organization and Architecture: Designing for Performance”, Eighth Edition, Pearson.
3. Dr. M. Usha, T. S. Srikanth, “Computer System Architecture and Organization”, First Edition, Wiley-India.
4. <https://www.gyaanibuddy.com/assignments/assignment-detail/fifo-page-replacement-algorithm/>

**---------------------------------------------------------------------------------------------------------**

**Pre Lab/ Prior Concepts:**

The FIFO algorithm uses the principle that the block in the set which has been in the block for the longest time is replaced. FIFO is easily implemented as a round robin or criteria buffer technique. The data structure used for implementation is a queue. Assume that the number of cache pages is three. Let the request to this cache is shown alongside.

**Algorithm:**

1. A hit is said to be occurred when a memory location requested is already in the cache.

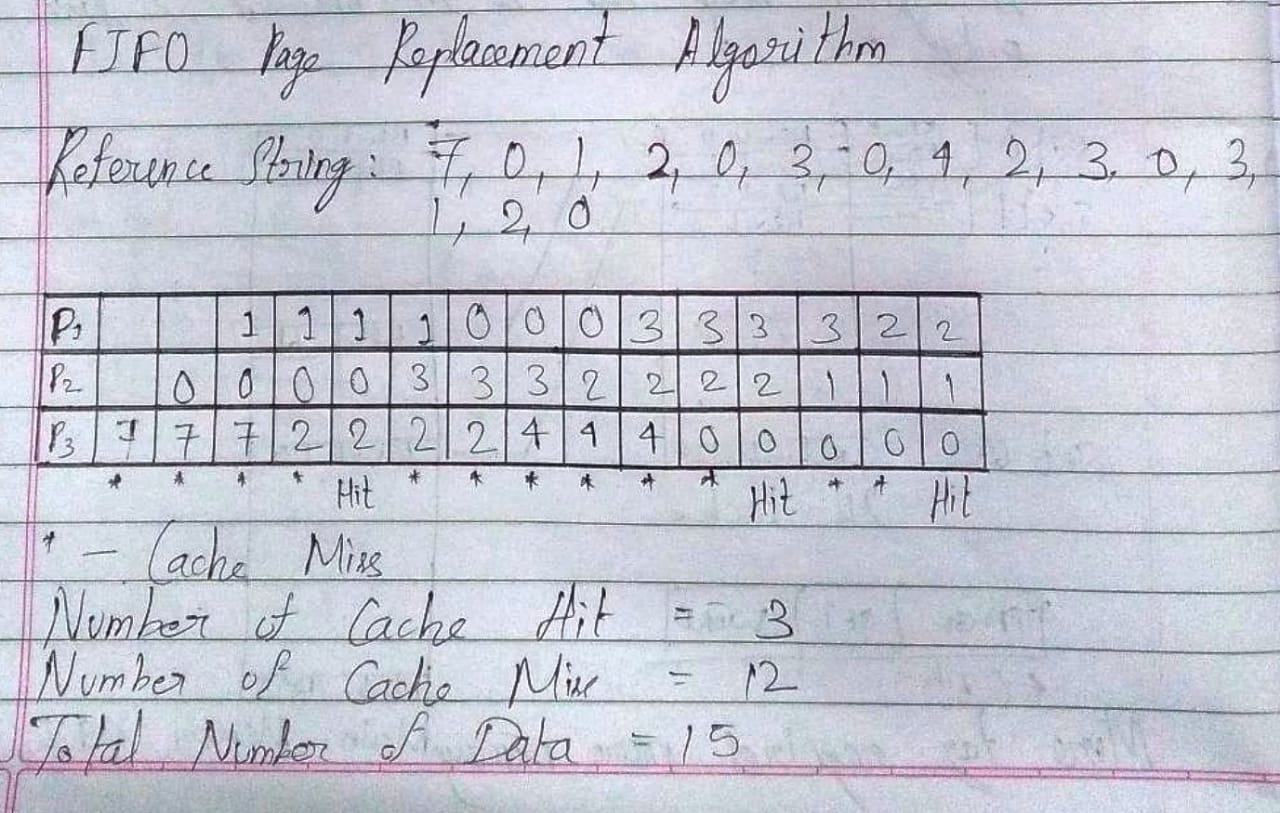
2. When cache is not full, the number of blocks is added.

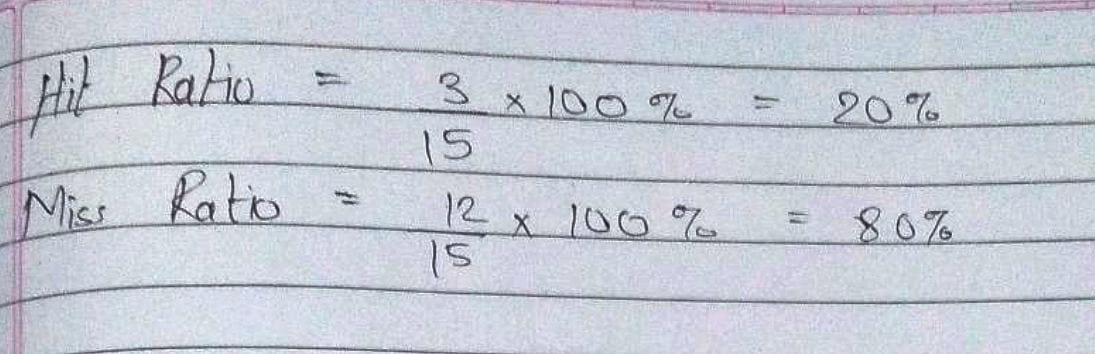
3. When cache is full, the block is replaced which was added first

**Design Steps:**

1. Start
2. Get input as memory block to be added to cache
3. Consider an element of the array
4. If cache is not full, add element to the cache array
5. If cache is full, check if element is already present
6. If it is hit is incremented
7. If not, element is added to cache removing first element (which is in first).
8. Repeat step 3 to 7 for remaining elements
9. Display the cache at very instance of step 8
10. Print hit ratio
11. End.

**Example:**





Code:

#include<stdio.h>

int main()

{

int i, j, k, num\_p, orig\_num\_p, pg[100], fr[10], num\_f, avail, miss, hit;

float miss\_ratio, hit\_ratio;

char ch;

do

{

miss = 0;

printf("\nEnter the number of pages (maximum 100): ");

scanf("%d", &num\_p);

orig\_num\_p = num\_p;

for(i = 1; i <= num\_p; i++)

{

printf("\nFor page %d, enter data: ", i);

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

}

printf("\nEnter the number of frames (maximum 10): ");

scanf("%d", &num\_f);

for(i = 0; i < num\_f; i++)

fr[i] = -1;

j = 0;

printf("\nReference String\tPage Frame\n");

for(i = 1; i <= num\_p; i++)

{

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

avail = 0;

for(k = 0; k < num\_f; k++)

if(fr[k] == pg[i])

avail = 1;

if(avail == 0)

{

fr[j] = pg[i];

j = (j+1)%num\_f;

miss++;

for(k = 0; k < num\_f; k++)

printf("%d\t", fr[k]);

}

printf("\n");

}

printf("\nThe number of Cache Miss is %d.", miss);

hit = orig\_num\_p - miss;

printf("\nThe number of Cache Hit is %d.", hit);

miss\_ratio = (miss\*100)/orig\_num\_p;

printf("\nThe Miss Ratio is %f.", miss\_ratio);

hit\_ratio = 100 - miss\_ratio;

printf("\nThe Hit Ratio is %f.", hit\_ratio);

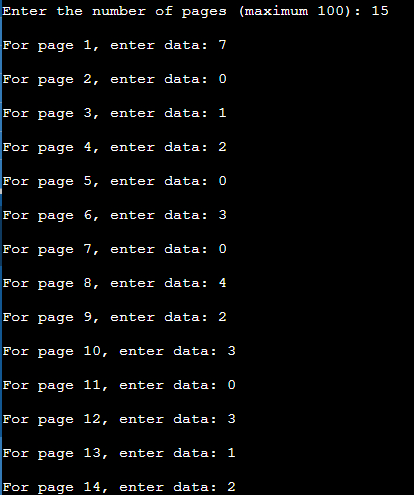
printf("\nDo you want to try again?\nIf so, then press 'Y' otherwise press any other character.\nEnter your choice: ");

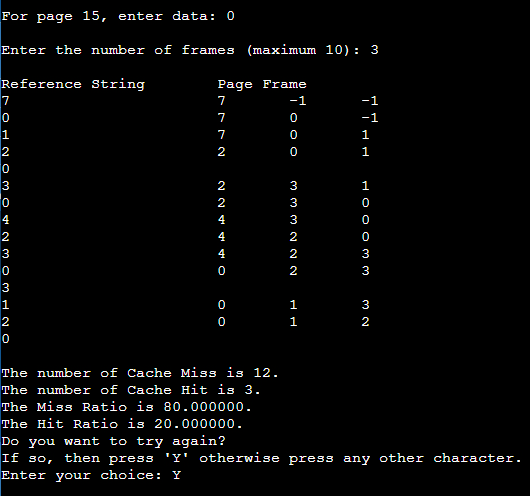
scanf("%s", &ch);

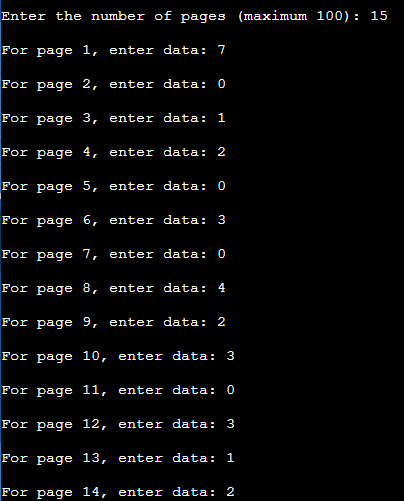
}while(ch=='Y');

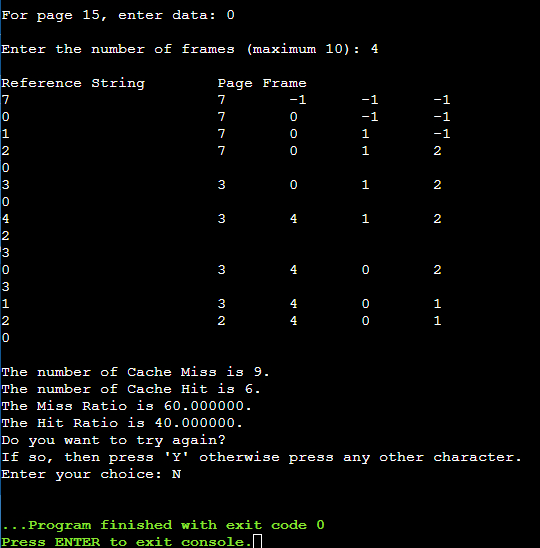
}

Output:









**Post Lab Descriptive Questions**

1. **What is meant by memory interleaving?**

**Ans.** Memory interleaving is a technique for compensating the relatively slow speed of DRAM (Dynamic RAM). In this technique, the main memory is divided into memory banks which can be accessed individually without any dependency on the other. For example, if there are four memory banks (four-way Interleaved Memory), with each containing 256 bytes, then, the Block Oriented scheme (no interleaving), will assign virtual address 0 to 255 to the first bank, 256 to 511 to the second bank. But in interleaved memory, virtual address 0 will be with the first bank, 1 with the second memory bank, 2 with the third bank and 3 with the fourth, and then 4 with the first memory bank again. Hence, the CPU can access alternate sections immediately without waiting for memory to be cached. There are multiple memory banks which take turns for supply of data. Memory interleaving is a technique for increasing memory speed. It is a process that makes the system more efficient, fast and reliable.

1. **Explain Paging Concept?**

Ans. Paging is a function of memory management where a computer will store and retrieve data from a device’s secondary storage to the primary storage. Memory management is a crucial aspect of any computing device, and paging specifically is important for the implementation of virtual memory.

Paging works by writing data to, and reading it from, secondary storage for use in primary storage. Paging is a basic function in memory management for a computer’s operating system (OS) as well – this includes Windows, Unix, Linux and Mac OS.

In a memory management system that takes advantage of paging, the OS reads data from secondary storage in blocks called pages, all of which have identical size. The physical region of memory containing a single page is called a frame. When paging is used, a frame does not have to comprise a single physically contiguous region in secondary storage. This approach offers an advantage over earlier memory management methods, because it facilitates more efficient and faster use of storage.

**Conclusion**

Thus, in this experiment, the concept of FIFO Page Replacement Algorithm has been learnt and implemented in C language. FIFO Page Replacement is part of the concept known as Paging, which is used in the memory management system of a computer to improve the efficiency of inserting and reading data items in memory.

**Date: \_\_17-11-22\_\_ Signature of faculty in-charge**