

LABWORK 5

Objective: Write C programs which implement following page replacement policies.

- a) First In First Out (FIFO)
- b) Least Recently Used (LRU)

Description: In multiprogramming system using dynamic partitioning there will come a time when all of the processes in the main memory are in a blocked state and there is insufficient memory. To avoid wasting processor time waiting for an active process to become unblocked. The OS will swap one of the process out of the main memory to make room for a new process or for a process in Ready- Suspend state. Therefore, the OS must choose which process to replace. Thus, when a page fault occurs, the OS has to change a page to remove from memory to make room for the page that must be brought in. If the page to be removed has been modified while in memory it must be written to disk to bring the disk copy up to date. Replacement algorithms can affect the system's performance.

Least Recently Used Algorithm

This paging algorithm selects a page for replacement that has been unused for the longest time.

First-In-First_Out

Replace the page that has been in memory longest, is the policy applied by FIFO. Pages from memory are removed in round-robin fashion. Its advantage is it's simplicity.

FIFO Sample Code:

```
#include<stdio.h>

#include<conio.h>

void main()
{
    int a[5],b[20],n,p=0,q=0,m=0,h,k,i,q1=1; char f='F';
    printf("Enter the Number of Pages:");
    scanf("%d",&n);
    printf("Enter %d Page Numbers:",n);
    for(i=0;i<n;i++)
        scanf("%d",&b[i]);

    for(i=0;i<n;i++){
        if(p==0){
            if(q>=3)
                q=0;
            a[q]=b[i];
```

```

        q++;
        if(q1<3){
            q1=q;
        }
    }
    printf("\n%d",b[i]);
    printf("\t");
    for(h=0;h<q1;h++)
        printf("%d",a[h]);
    if((p==0)&&(q<=3)){
        printf("-->%c",f);m++;
    }
    p=0;
    for(k=0;k<q1;k++){
        if(b[i+1]==a[k])
            p=1;
    }
}
printf("\nNo of faults:%d",m); getch();
}

```

TASK: Write C program which implements the Least Recently Used (LRU) page replacement algorithm.

OUTPUT:

```

Enter the number of pages:12
Enter 12 Page Numbers:7 3 7 1 5 7 4 5 3 7 5 7
7      7-->F
3      73-->F
7      73
1      731-->F
5      751-->F
7      751
4      754-->F
5      754
3      354-->F
7      357-->F
5      357
7      357
No of faults:7

```