

**Operating System – CS23431**

|                             |   |
|-----------------------------|---|
| <b>Ex 11c)</b>              | <b>Optimal Page Replacement Algorithm</b> |
| <b>Name: B M Madhumitha</b> |   |
| <b>Reg No: 230701168</b>    |   |

Aim:

To write a c program to implement Optimal page replacement algorithm.

ALGORITHM:

1. Start the process
2. Declare the size
3. Get the number of pages to be inserted
4. Get the value
5. Declare counter and stack
6. Select the least frequently used page by counter value
7. Stack them according the selection.
8. Display the values
9. Stop the process

**PROGRAM:**

```
#include<stdio.h>

int predict(int frames[],int frames_occ, int n, int mem[],int index){
    int farthest = index,result = -1;
    for(int j=0;j<frames_occ;j++){
        int i;
        for(i=index;i<n;i++){
            if(frames[j]==mem[i]){
                if(i>farthest){
                    farthest = i;
                    result = j;
                }
            }
        }
        break;
    }
}
```

```

        if(i == n)return j;
    }

    return result;
}
int search(int frames[],int size,int key){
    for(int i=0;i<size;i++){
        if(frames[i]==key)return i;
    }
    return -1;
}
void display(int size,int frames[]){
    for(int i=0;i<size;i++){
        if(frames[i]==-1)printf("- ");
        else printf("%d ",frames[i]);

    }
    printf("\n");
}
void optimalreplacement(int n, int size, int frames[],int mem[]){
    int frame_occupied=0,page_fault=0,hits = 0,found=-1,flag=0;
    printf("Frames\n");
    for(int i=0;i<n;i++){
        flag=0;
        found = search(frames,size,mem[i]);
        if(frame_occupied<size){
            if(found==-1){
                frames[frame_occupied]=mem[i];
                frame_occupied++;
                page_fault++;
            } else{
                hits++;
                flag = 1;
            }
        }
        else{
            if(found==-1){

                int min_idx = predict(frames,size,n,mem,i);
                frames[min_idx]=mem[i];
                page_fault++;
            } else{
                hits++;
                flag=1;
            }
        }
        if(flag == 1){
            printf("There is no Page Fault!!\n");
        } else{
            display(size,frames);
        }
    }
}

```

```

    }
}
printf("\nThe total no. of page fault:%d",page_fault);
printf("\nThe total no. of Hits:%d",hits);
}
int main(){
    int size,n;
    printf("Enter the no. of frames:");
    scanf("%d",&size);
    int frames[size];
    for(int i=0;i<size;i++){
        frames[i]=-1;
    }
    printf("Enter the no. of values:");
    scanf("%d",&n);
    int mem[n];
    printf("Enter the values:\n");
    for(int i=0;i<n;i++){
        printf("Enter val%d:",i+1);
        scanf("%d",&mem[i]);
    }
    optimalreplacement(n,size,frames,mem);
}

```

Output:

```

C:\Users\kambm\OneDrive\Desktop\Madhumitha\sem IV\OS Assignment\Final version>gcc optimal.c -o optimal.exe
C:\Users\kambm\OneDrive\Desktop\Madhumitha\sem IV\OS Assignment\Final version>optimal.exe
Enter the no. of frames:3
Enter the no. of values:10
Enter the values:
Enter val1:2
Enter val2:0
Enter val3:1
Enter val4:5
Enter val5:6
Enter val6:2
Enter val7:4
Enter val8:0
Enter val9:3
Enter val10:1
Frames
2 - -
2 0 -
2 0 1
2 0 5
2 0 6
There is no Page Fault!!
4 0 6
There is no Page Fault!!
3 0 6
1 0 6

The total no. of page fault:8
The total no. of Hits:2
C:\Users\kambm\OneDrive\Desktop\Madhumitha\sem IV\OS Assignment\Final version>

```

Result: Thus, the Program was executed successfully