

OPERATING SYSTEM - CS23431

EXP 11(c)

OPTIMAL

NAME: KEERTHANA.S

ROLL NO: 230701149

PROGRAM:

```
#include <stdio.h>
int findreplacementindex(int n,int frame_size,int page[],int mem[],int current)
{
    int ind[frame_size];
    for(int i=0;i<frame_size;i++)
    {
        ind[i]=-1;
        for(int j=current+1;j<n;j++)
        {
            if(mem[i]==page[j])
            {
                ind[i]=j;
                break;
            }
        }
    }
    int dist=-1,reqind=-1;;
    for(int i=0;i<frame_size;i++)
    {
        if(ind[i]==-1)
        {
            return i;
        }
        else if(ind[i]>dist)
        {
            dist=ind[i];
            reqind=i;
        }
    }
    return reqind;
}
int main() {
    int n,frame_size,front=0,count=0,page_faults=0;
    printf("Enter size of reference string: ");
    scanf("%d",&n);
```

```

int page[n];
for(int i=0;i<n;i++)
{
    printf("Enter [%d]: ",i+1);
    scanf("%d",&page[i]);
}
printf("Enter page frame size: ");
scanf("%d",&frame_size);
int mem[frame_size];
for (int i = 0; i < n; i++) {
    int found = 0;
    for (int j = 0; j < count; j++) {
        if (mem[j] == page[i]) {
            found = 1;
            break;
        }
    }
}

printf("%d -> ", page[i]);
int f=1;
if (!found) {
    if (count < frame_size) {
        mem[count++] = page[i];
    } else {
        int index=findreplacementindex(n,frame_size,page,mem,i);
        mem[index]=page[i];
    }
    page_faults++;
}
else
{
    f=0;
    printf("No Page Fault ");
}
if(f)
{
    for (int j = 0; j < count; j++) {
        printf("%d ", mem[j]);
    }
}

printf("\n");
}

```

```
printf("\nTotal Page Faults: %d\n", page_faults);

return 0;
}
```

OUTPUT:

```
Enter size of reference string: 7
Enter [1]: 7
Enter [2]: 0
Enter [3]: 1
Enter [4]: 2
Enter [5]: 0
Enter [6]: 3
Enter [7]: 0
Enter page frame size: 3
7 -> 7
0 -> 7 0
1 -> 7 0 1
2 -> 2 0 1
0 -> No Page Fault
3 -> 3 0 1
0 -> No Page Fault

Total Page Faults: 5
```