# OPERATING SYSTEM - CS23431 EXP 11(C)

**OPTIMAL PAGE REPLACEMENT**

**NAME: KAAVIYA.R ROLL NO: 230701504 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:

