//lru algorithm

#include<stdio.h>

int findLRU(int time[], int n)

{

int i,minimum=time[0],pos=0;

for(i=1;i<n;i++)

{

if(time[i]<minimum)

{

minimum=time[i];

pos=i;

}

}

return pos;

}

int main()

{

int num,n,frames[10],pages[30],counter=0,time[10],flag1,flag2,i,j,pos,faults=0;

printf("Enter number of frames: ");

scanf("%d", &num);

printf("Enter number of pages: ");

scanf("%d", &n);

printf("Enter reference string: ");

for(i=0;i<n;i++)

{

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

}

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

{

frames[i]=-1;

}

for(i=0;i<n;i++)

{

flag1 = flag2 = 0;

for(j=0;j<num;j++)

{

if(frames[j]==pages[i])

{

counter++;

time[j] = counter;

flag1 = flag2 = 1;

break;

}

}

if(flag1==0)

{

for(j=0;j<num;j++)

{

if(frames[j]==-1)

{

counter++;

faults++;

frames[j] = pages[i];

time[j] = counter;

flag2 = 1;

break;

}

}

}

if(flag2==0)

{

pos = findLRU(time,num);

counter++;

faults++;

frames[pos] = pages[i];

time[pos] = counter;

}

printf("\n");

for(j=0;j<num;j++)

{

printf("%d\t", frames[j]);

}

}

printf("\n\nTotal Page Faults = %d", faults);

return 0;

}

Output:

