**EXPERIMENT N0-9**

**Write a program to implement page replacement policies: LRU**

**CODE:**

#include<stdio.h>

int x,x1;

int in[100],p[50],hit=0,i,j,k,pgfaultcnt=0;

void initialize()

{

pgfaultcnt=0;

for(i=0; i<x1; i++)

p[i]=9999;

}

int isHit(int data)

{

hit=0;

for(j=0; j<x1; j++)

{

if(p[j]==data)

{

hit=1;

break;

}

}

return hit;

}

int getHitIndex(int data)

{

int hitind;

for(k=0; k<x1; k++)

{

if(p[k]==data)

{

hitind=k;

break;

}

}

return hitind;

}

void dispPages()

{

for (k=0; k<x1; k++)

{

if(p[k]!=9999)

printf(" %d",p[k]);

}

}

void dispPgFaultCnt()

{

printf("\nTotal no of page faults:%d",pgfaultcnt);

}

void LRU()

{

initialize();

int least[50];

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

{

printf("\nFor %d :",in[i]);

if(isHit(in[i])==0)

{

for(j=0; j<x1; j++)

{

int pg=p[j];

int found=0;

for(k=i-1; k>=0; k--)

{

if(pg==in[k])

{

least[j]=k;

found=1;

break;

}

else

found=0;

}

if(!found)

least[j]=-9999;

}

int min=9999;

int repindex;

for(j=0; j<x1; j++)

{

if(least[j]<min)

{

min=least[j];

repindex=j;

}

}

p[repindex]=in[i];

pgfaultcnt++;

dispPages();

}

else

printf("No page fault!");

}

dispPgFaultCnt();

}

int main()

{

printf("\nEnter no of pages:");

scanf("%d",&x);

printf("\nEnter refernce string:");

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

{

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

}

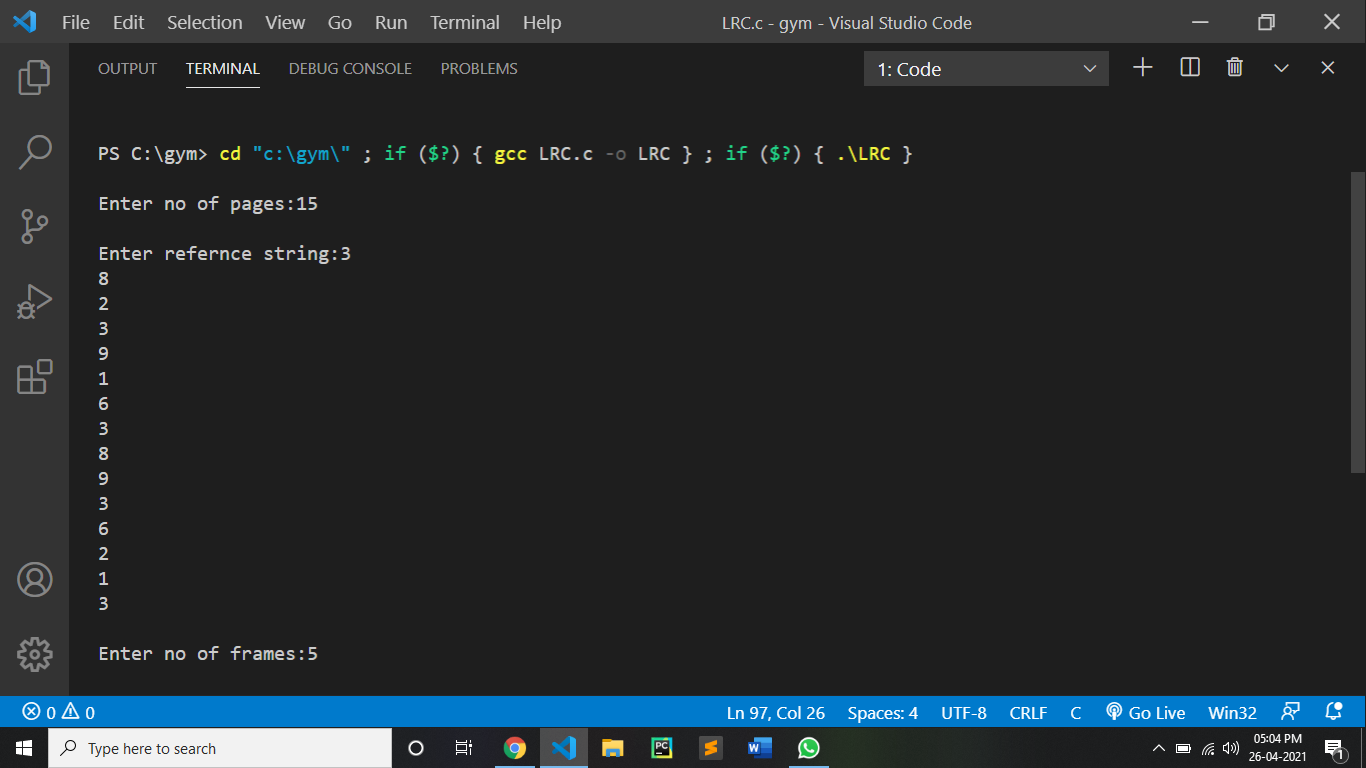
printf("\nEnter no of frames:");

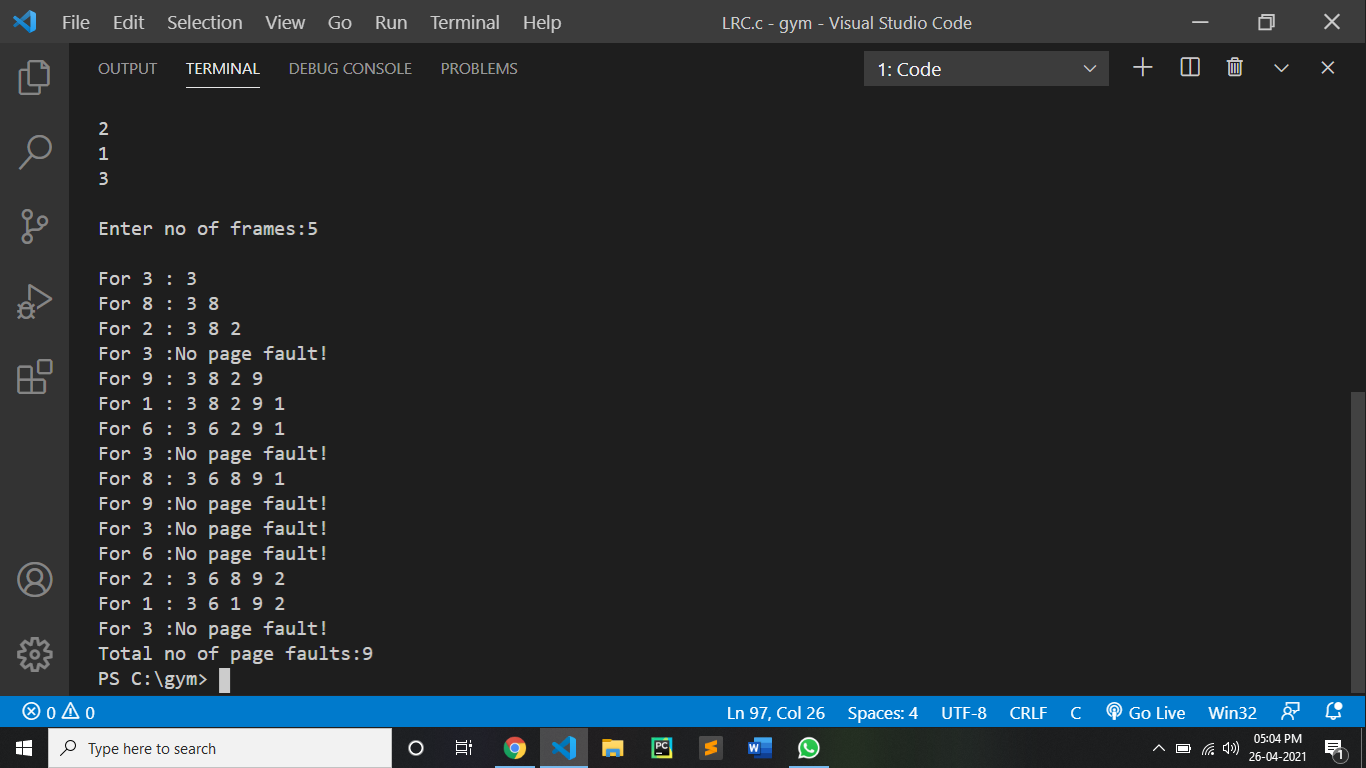
scanf("%d",&x1);

LRU();

}

**OUTPUT:**

****

****