**RSA:PROGRAM-08**

**rsa.c**

#include<stdio.h>

#include<stdlib.h>

#include<math.h>

#include<string.h>

long int p,q,n,t,flag,e[100],d[100],temp[100],j,m[100],en[100],i;

char msg[100];

int prime(long int);

void ce();

long int cd(long int);

void encrypt();

void decrypt();

void main() {

printf("\nENTER FIRST PRIME NUMBER\n");

scanf("%d",&p);

flag=prime(p);

if(flag==0) {

printf("\nWRONG INPUT\n");

exit(1);

}

printf("\nENTER ANOTHER PRIME NUMBER\n");

scanf("%d",&q);

flag=prime(q);

if(flag==0||p==q) {

printf("\nWRONG INPUT\n");

exit(1);

}

printf("\nENTER MESSAGE\n");

fflush(stdin);

scanf("%s",msg);

for (i=0;msg[i]!=NULL;i++)

m[i]=msg[i];

n=p\*q;

t=(p-1)\*(q-1);

ce();

printf("\nPOSSIBLE VALUES OF e AND d ARE\n");

for (i=0;i<j-1;i++)

printf("\n%ld\t%ld",e[i],d[i]);

encrypt();

decrypt();

}

int prime(long int pr) {

int i;

j=sqrt(pr);

for (i=2;i<=j;i++) {

if(pr%i==0)

return 0;

}

return 1;

}

void ce() {

int k;

k=0;

for (i=2;i<t;i++) {

if(t%i==0)

continue;

flag=prime(i);

if(flag==1&&i!=p&&i!=q) {

e[k]=i;

flag=cd(e[k]);

if(flag>0) {

d[k]=flag;

k++;

}

if(k==99)

break;

}

}

}

long int cd(long int x) {

long int k=1;

while(1) {

k=k+t;

if(k%x==0)

return(k/x);

}

}

void encrypt() {

long int pt,ct,key=e[0],k,len;

i=0;

len=strlen(msg);

while(i!=len) {

pt=m[i];

pt=pt-96;

k=1;

for (j=0;j<key;j++) {

k=k\*pt;

k=k%n;

}

temp[i]=k;

ct=k+96;

en[i]=ct;

i++;

}

en[i]=-1;

printf("\nTHE ENCRYPTED MESSAGE IS\n");

for (i=0;en[i]!=-1;i++)

printf("%c",en[i]);

}

void decrypt() {

long int pt,ct,key=d[0],k;

i=0;

while(en[i]!=-1) {

ct=temp[i];

k=1;

for (j=0;j<key;j++) {

k=k\*ct;

k=k%n;

}

pt=k+96;

m[i]=pt;

i++;

}

m[i]=-1;

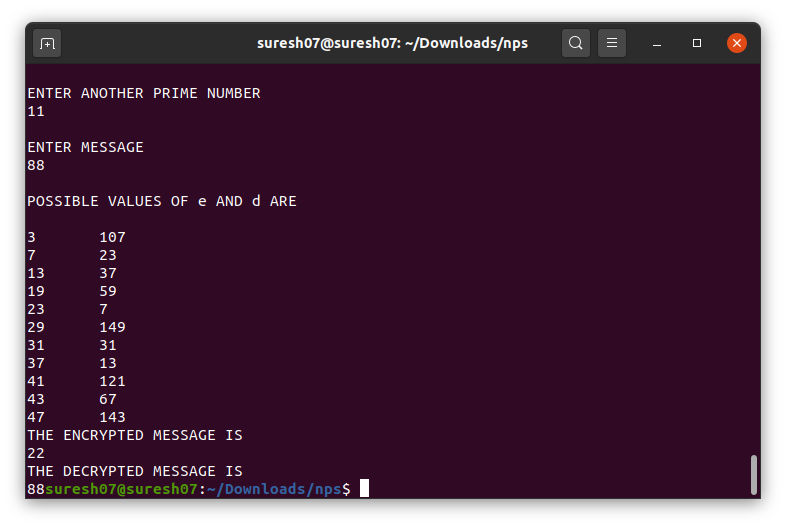
printf("\nTHE DECRYPTED MESSAGE IS\n");

for (i=0;m[i]!=-1;i++)

printf("%c",m[i]);

}

**OUTPUT**

****