```
//C++ program for encryption and decryption RSA algorithm

#include<iostream>

#include<stdlib.h>

#include<math.h>

#include<string.h>


using namespace std;


int x, y, n, t, i, flag;

long int e[50], d[50], temp[50], j;

char en[50], m[50];

char msg[100];

int prime(long int); //function to check for prime number

void encryption_key();

long int cd(long int);

void encrypt();

void decrypt();


int main()

{

  cout << "\nENTER FIRST PRIME NUMBER\n";

  cin >> x;


  //checking whether input is prime or not

  flag = prime(x);
```

```
if(flag == 0)
{
   cout << "\nINVALID INPUT\n";
   exit(0);
}

cout << "\nENTER SECOND PRIME NUMBER\n";
cin >> y;

flag = prime(y);
if(flag == 0 || x == y)
{
   cout << "\nINVALID INPUT\n";
   exit(0);
}

cout << "\nENTER MESSAGE OR STRING TO ENCRYPT\n";
cin >> msg;

for(i = 0; msg[i] != NULL; i++)
   m[i] = msg[i];
n = x * y;
t = (x - 1) * (y - 1);

encryption_key();
```

```
  cout << "\nPOSSIBLE VALUES OF e AND d ARE\n";

  for(i = 0; i < j - 1; i++)
    cout << "\n" << e[i] << "\t" << d[i];

  encrypt();
  decrypt();
  return 0;
} //end of the main program

int prime(long int pr)
{
  int i;
  j = sqrt(pr);
  for(i = 2; i <= j; i++)
  {
    if(pr % i == 0)
      return 0;
  }
  return 1;
 }

//function to generate encryption key
void encryption_key()
{
```

```
  int k;
  k = 0;
  for(i = 2; i < t; i++)
  {
    if(t % i == 0)
      continue;
    flag = prime(i);
    if(flag == 1 && i != x && i != y)
    {
      e[k] = i;
      flag = cd(e[k]);
      if(flag > 0)
      {
        d[k] = flag;
        k++;
      }
      if(k == 99)
      break;
    }
  }
}

long int cd(long int a)
{
  long int k = 1;
```

```
  while(1)
  {
    k = k + t;
    if(k % a == 0)
      return(k/a);
  }
}

//function to encrypt the message
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;
    cout << "\n\nTHE ENCRYPTED MESSAGE IS\n";
    for(i=0; en[i] != -1; i++)
      cout << en[i];
}

//function to decrypt the message
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;

  cout << "\n\nTHE DECRYPTED MESSAGE IS\n";

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

    cout << m[i];



 cout << endl;

}
```

OUTPUT: