Date:11.11.2022

**Final Year B. Tech., Sem VII 2022-23**

**Cryptography And Network Security Lab**

**Assignment submission**

**PRN: 2019BTECS00071**

**Name: Charpale Abhishek**

**Batch: B3**

**Assignment: 12**

**Assignment 12: Implementation of RSA**

**Title:**

Implementation of RSA (Rivest–Shamir–Adleman)

**Aim:**

To develop and implement the RSA (Rivest–Shamir–Adleman)

**Theory:**

* RSA (Rivest–Shamir–Adleman) is a public-key cryptosystem that is widely used for secure data transmission. It is also one of the oldest.
* The acronym "RSA" comes from the surnames of Ron Rivest, Adi Shamir and Leonard Adleman, who publicly described the algorithm in 1977. An equivalent system was developed secretly in 1973 at GCHQ (the British signals intelligence agency) by the English mathematician Clifford Cocks.
* An RSA user creates and publishes a public key based on two large prime numbers, along with an auxiliary value. The prime numbers are kept secret. Messages can be encrypted by anyone, via the public key, but can only be decoded by someone who knows the prime numbers.
* The security of RSA relies on the practical difficulty of factoring the product of two large prime numbers, the "factoring problem".

**Implementation of RSA (Rivest–Shamir–Adleman)**

**Code:**

/\*

\* C++ Program to Implement the RSA Algorithm

\*/

#include<iostream>

#include<math.h>

#include<string.h>

#include<stdlib.h>

using namespace std;

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 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;

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

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

{

printf("%c", (int)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;

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

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

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

}

int prime(long int pr)

{

int i;

j = sqrt(pr);

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

{

if (pr % i == 0)

return 0;

}

return 1;

}

int main()

{

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

cin >> p;

flag = prime(p);

if (flag == 0)

{

cout << "\nWRONG INPUT\n";

exit(1);

}

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

cin >> q;

flag = prime(q);

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

{

cout << "\nWRONG INPUT\n";

exit(1);

}

cout << "\nENTER MESSAGE\n";

fflush(stdin);

cin >> msg;

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

m[i] = msg[i];

n = p \* q;

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

ce();

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

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

cout << e[i] << "\t" << d[i] << "\n";

encrypt();

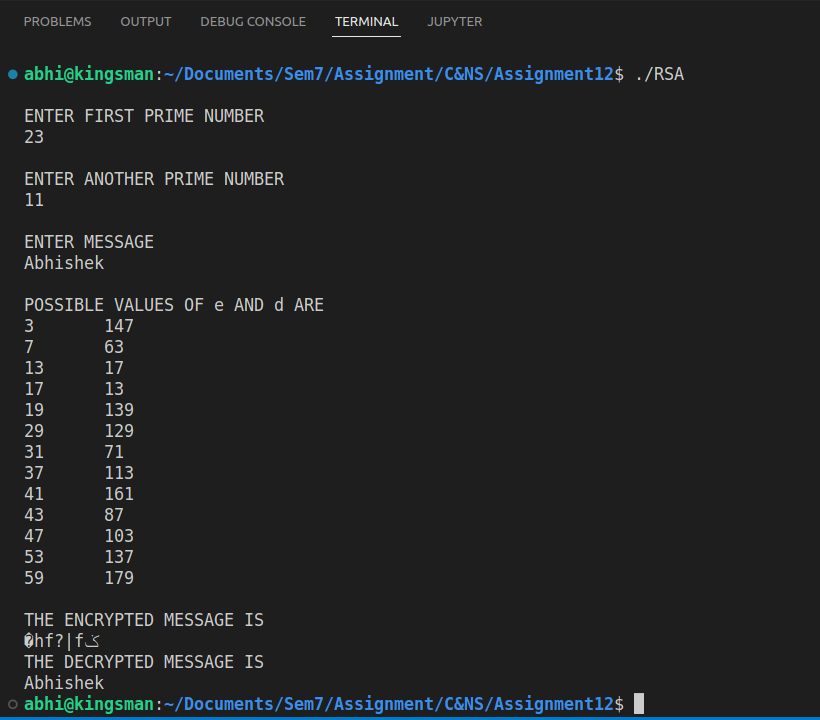
decrypt();

cout<<"\n";

return 0;

}

**Output:**



**Conclusion:**

Performed the experiment successfully.

The RSA can be used to do data transmission