Name : Harshal Kodgire
PRN : 2019BTECS00029
Batch : B1
Topic : CNS Assignment 12

Aim : To Implement RSA algorithm

Theory : RSA algorithm is an asymmetric cryptography algorithm. Asymmetric actually means that it works on two different keys i.e. Public Key and Private Key. As the name describes, the Public Key is given to everyone and the Private key is kept private. An example of asymmetric cryptography : A client (for example browser) sends its public key to the server and requests some data. The server encrypts the data using the client's public key and sends the encrypted data. The client receives this data and decrypts it.

Code :

```
#include <bits/stdc++.h>
using namespace std;

// Function for extended Euclidean Algorithm
int ansS, ansT;
int findGcdExtended(int r1, int r2, int s1, int s2, int t1, int t2)
{
    // Base Case
    if (r2 == 0)
    {
        ansS = s1;
        ansT = t1;
        return r1;
    }

    int q = r1 / r2;
    int r = r1 % r2;

    int s = s1 - q * s2;
    int t = t1 - q * t2;

    cout << q << " " << r1 << " " << r2 << " " << r << " " << s1 << " "
<< s2 << " " << s << " " << t1 << " " << t2 << " " << t << endl;

    return findGcdExtended(r2, r, s2, s, t2, t);
}
```

```
int modInverse(int A, int M)
{
    int x, y;
    int g = findGcdExtended(A, M, 1, 0, 0, 1);
    if (g != 1) {
        cout << "\n Inverse doesn't exist";
        return 0;
    }
    else {

        // m is added to handle negative x

        int res = (ansS % M + M) % M;
        cout << "\n Inverse is" << res << endl;
        return res;
    }
}

long long powM(long long a, long long b, long long n)
{
    if (b == 1)
        return a % n;
    long long x = powM(a, b / 2, n);
    x = (x * x) % n;
    if (b % 2)
        x = (x * a) % n;
    return x;
}

int findGCD(int num1, int num2)
{
    if (num1 == 0)
        return num2;
    return findGCD(num2 % num1, num1);
}

// Code to demonstrate RSA algorithm
int main()
{

    // Two random prime numbers
    long long p, q, e, msg;
```

```
    //17 31 7 2

    cout << "\n Please enter 2 prime number : ";
    cin >> p >> q;

    cout << "\n Enter value of e : ";
    cin >> e;

    cout << "\n Enter message to encrpyt : ";
    cin >> msg;

    cout << "\n 2 random prime numbers selected are " << p << " " << q
<< endl;

    // First part of public key:
    long long n = p * q;
    cout << "\n Product of two prime number n is " << n << endl;

    // Finding other part of public key.
    // e stands for encrypt

    cout << "\n Taken e is " << e << endl;

    long long phi = (p - 1) * (q - 1);
    cout << "\n phi is " << phi << endl;

    while (e < phi) {
        // e must be co-prime to phi and
        // smaller than phi.
        if (findGCD(e, phi) == 1)
            break;
        else
            e++;
    }

    cout << "\n Final e value is " << e << endl;


    // Private key (d stands for decrypt)

    long long d = modInverse(e, phi);
    cout << "\n d is " << d << endl;
```

```
    cout << "\n So now our public key is " << "<" << e << "," << n <<
">" << endl;
    cout << "\n So now our private key is " << "<" << d << "," << n <<
">" << endl << endl;

    // Message to be encrypted

    cout << "\n Message date is " << msg << endl;

    // Encryption c = (msg ^ e) % n
    long long c = powM(msg, e, n);
    cout << "\n Encrypted Message is " << c << endl;

    // Decryption m = (c ^ d) % n
    long long m = powM(c, d, n);
    cout << "\n Original Message is " << m << endl;

    return 0;
}
```

Output :

```
D:\WCE_ENGINEERING\BTECH_SEM1\CNS lab\LA2>g++ assignment11_rsa.cpp

D:\WCE_ENGINEERING\BTECH_SEM1\CNS lab\LA2>a.exe

 Please enter 2 prime number : 3
11

 Enter value of e : 7

 Enter message to encrpyt : 40

 2 random prime numbers selected are 3 11

 Product of two prime number n is 33

 Taken e is 7

 phi is 20

 Final e value is 7
0 7 20 7 1 0 1 0 1 0
2 20 7 6 0 1 -2 1 0 1
1 7 6 1 1 -2 3 0 1 -1
6 6 1 0 -2 3 -20 1 -1 7

 Inverse is3

 d is 3

 So now our public key is <7,33>

 So now our private key is <3,33>


 Message date is 40

 Encrypted Message is 28

 Original Message is 7

D:\WCE_ENGINEERING\BTECH_SEM1\CNS lab\LA2>
```