Cryptography and Information Security LAB

Practical-4

Name:- Mohit Nehete

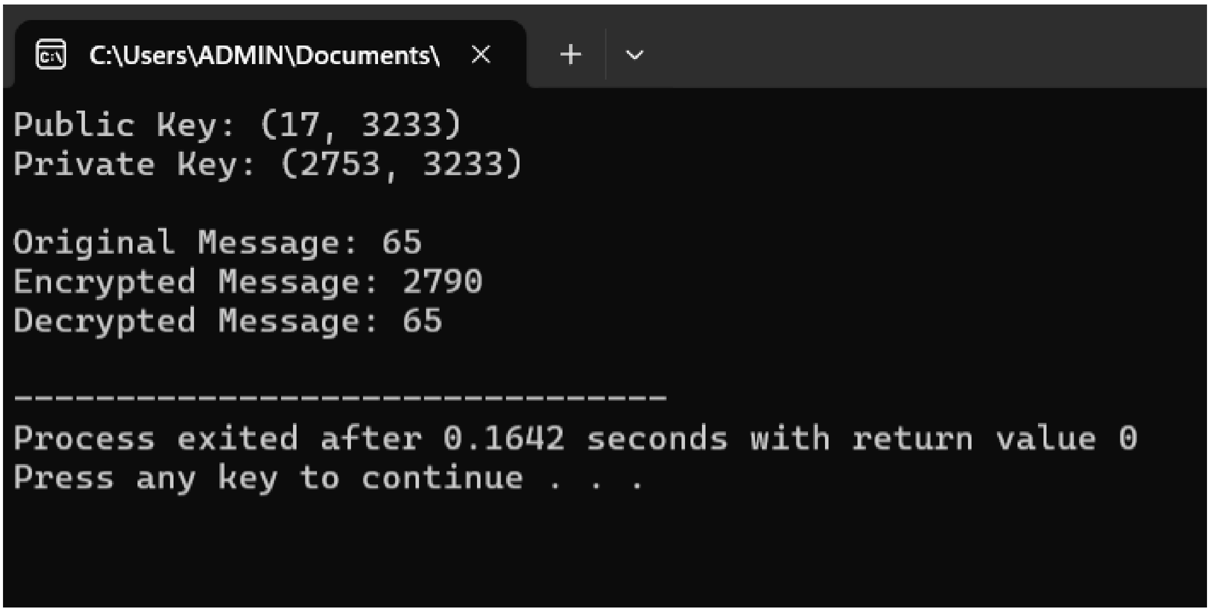
PRN:- 202301040201

Question:- **Write a program to implement public key cryptography algorithm RSA.**

Code:-

```
#include <iostream>
#include <cmath>
using namespace std;

int gcd(int a, int b) {
    if (b == 0)
        return a;
    return gcd(b, a % b);
}
int modInverse(int e, int phi) {
    int d = 0, x1 = 0, x2 = 1, y1 = 1, tempPhi = phi;
    while (e > 0) {
        int temp1 = tempPhi / e;
        int temp2 = tempPhi - temp1 * e;
        tempPhi = e;
        e = temp2;

        int x = x2 - temp1 * x1;
        int y = d - temp1 * y1;

        x2 = x1;
        x1 = x;
        d = y1;
        y1 = y;
    }
    if (tempPhi == 1)
        return (d + phi) % phi;
    return -1;
}

long long power(long long m, long long k, long long n) {
    long long result = 1;
    m = m % n;
    while (k > 0) {
        if (k % 2 == 1)
            result = (result * m) % n;
        k = k / 2;
        m = (m * m) % n;
    }
    return result;
}

int main() {
    int p = 61;
    int q = 53;
    int n = p * q;
    int phi = (p - 1) * (q - 1);

    int e = 17;
    if (gcd(e, phi) != 1) {
        cout << "e and phi are not coprime!" << endl;
        return 0;
    }
    int d = modInverse(e, phi);

    cout << "Public Key: (" << e << ", " << n << ")" << endl;
    cout << "Private Key: (" << d << ", " << n << ")" << endl;

    long long msg = 65;
    cout << "\nOriginal Message: " << msg << endl;

    long long cipher = power(msg, e, n);
    cout << "Encrypted Message: " << cipher << endl;

    long long decrypted = power(cipher, d, n);
    cout << "Decrypted Message: " << decrypted << endl;

    return 0;
}
```

Output:-



```
Public Key: (17, 3233)
Private Key: (2753, 3233)

Original Message: 65
Encrypted Message: 2790
Decrypted Message: 65

--------------------------------
Process exited after 0.1642 seconds with return value 0
Press any key to continue . . .
```