**Молдавский государственный университет**

**Факультет математики и информатики**

**Департамент Информатики**

**УПРАЖНЕНИЕ**

**RSA**

**Выполнил**: Гуцу Даниил.

**Группа:** IA2102

**Проверила:** др.,Чербу О.

Кишинев, 2023

**КИШИНЕВ – 2023**

Chisinau 2022

Алгоритм

* Выбираю два простых числа.
* Вычисляем модуль — произведение наших p и q
* Вычисляем функцию Эйлера: φ=(p-1)×(q-1).
* Выбираем число e, отвечающее следующим критериям: (i) оно должно быть простое, (ii) оно должно быть меньше φ.  Это открытая экспонента.
* Далее вычисляется число d, d = e^(-1) mod(Φ(n))

Шифрование

Шифрование выполняется по следующей формуле me (mod n)

Расшифровка

Расшифровка выполняется по следующей формуле med mod n

Код программы:

package individ1;  
  
import java.io.DataInputStream;  
import java.io.IOException;  
import java.math.BigInteger;  
import java.util.Random;  
  
public class RSA {  
 private final BigInteger N;  
 private final BigInteger e;  
 private final BigInteger d;  
  
 public RSA() {  
 Random r = new Random();  
 int maxLength = 1024;  
 BigInteger p = BigInteger.*probablePrime*(maxLength, r);  
 BigInteger q = BigInteger.*probablePrime*(maxLength, r);  
 N = p.multiply(q);  
 BigInteger PHI = p.subtract(BigInteger.*ONE*).multiply(q.subtract(BigInteger.*ONE*));  
 e = BigInteger.*probablePrime*(maxLength / 2, r);  
 while (PHI.gcd(e).compareTo(BigInteger.*ONE*) > 0 && e.compareTo(PHI) < 0) {  
 e.add(BigInteger.*ONE*);  
 }  
 d = e.modInverse(PHI);  
 }  
  
 public static void main(String[] args) {  
 RSA rsa = new RSA();  
 DataInputStream input = new DataInputStream(System.*in*);  
 String inputString;  
 System.*out*.println("Enter message you wish to send:");  
 try {  
 inputString = input.readLine();  
 System.*out*.println("Encrypting the message: " + inputString);  
 System.*out*.println("The message in bytes is: "  
 + *bytesToString*(inputString.getBytes()));  
 // encryption  
 byte[] cipher = rsa.encryptMessage(inputString.getBytes());  
 // decryption  
 byte[] plain = rsa.decryptMessage(cipher);  
 System.*out*.println("Decrypting Bytes: " + *bytesToString*(plain));  
 System.*out*.println("Plain message is: " + new String(plain));  
 } catch (IOException ioException) {  
 ioException.printStackTrace();  
 }  
  
  
 }  
  
 private static String bytesToString(byte[] cipher) {  
 String temp = "";  
 for (byte b : cipher) {  
 temp += Byte.*toString*(b);  
 }  
 return temp;  
 }  
  
 public byte[] encryptMessage(byte[] message) {  
 return (new BigInteger(message)).modPow(e, N).toByteArray();  
 }  
  
 // Decrypting the message  
 public byte[] decryptMessage(byte[] message) {  
 return (new BigInteger(message)).modPow(d, N).toByteArray();  
 }  
  
  
}

Вывод программы:

Enter message you wish to send:

criptogg

Encrypting the message: criptogg

The message in bytes is: 99114105112116111103103

Decrypting Bytes: 99114105112116111103103

Plain message is: criptogg