1. **RSA Program Code –**

import java.util.\*;

class RSAcrypto {

public static void main(String args[]) {

Scanner sc = new Scanner(System.in);

int d = 0;

System.out.println("Enter two prime numbers: ");

int p = sc.nextInt();

int q = sc.nextInt();

int n = p \* q;

System.out.println("n =" + n);

int e = 0;

int pn = (p - 1) \* (q - 1);

search:{

for (int i = 2; i <= pn; i++) {

int r;

int j = i;

int k = pn;

while (k != j) {

if (k > j) k = k - j; else j = j - k;

}

if (k == 1) {

e = i;

break search;

}

}

}

System.out.println("e =" + e);

go:{

for (int i = 1; i <= pn; i++) {

int x = (e \* i) % pn;

if (x == 1) {

System.out.println("d =" + i);

System.out.println("The private key is (d) " + i);

d = i;

break go;

}

}

}

System.out.println("The public key is (n.e) " + n + ", " + e);

String t;

int c;

System.out.println("Enter plaintext: ");

t = sc.next();

int m = 0;

for (int i = 0; i < t.length(); i++) {

m += (int) t.charAt(i);

}

c = ((m) ^ e) % n;

System.out.println("The Encryted message is " + m);

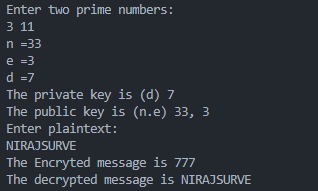
m = (c ^ d) % n;

System.out.println("The decrypted message is " + t);

}

}

**Output –**

****

1. **Digital Signature using RSA/EI Gamal Code –**

import java.security.KeyPair;

import java.security.KeyPairGenerator;

import java.security.Signature;

import java.util.Base64;

public class RSADigSig {

public static void main(String[] args) throws Exception {

KeyPairGenerator kpg = KeyPairGenerator.getInstance("RSA");

kpg.initialize(1024);

KeyPair keyPair = kpg.genKeyPair();

byte[] data = "NIRAJ".getBytes("UTF8");

Signature sig = Signature.getInstance("MD5WithRSA");

sig.initSign(keyPair.getPrivate());

sig.update(data);

byte[] signatureBytes = sig.sign();

System.out.println(

"Signature: " + Base64.getEncoder().encodeToString(signatureBytes));

sig.initVerify(keyPair.getPublic());

sig.update(data);

System.out.println(sig.verify(signatureBytes));

}

}

**Output –**

****