**RSA ALGORITHM**

Ex.NO -06 **. T.HARSHINI**

15-09-20 **(2017103532)**

**AIM –**

In this exercise , we will be implementing RSA algorithm for cryptographic study.

**RSA ALOGRITHM -**

An asymmetric cryptography algorithm. Asymmetric means that it works on two different keys i.e. **Public Key** and **Private Key.** As the name suggests that the **Public Key** is given to everyone and **Private Key** is kept private.

**ALGORITHM:**

Step 1 : Choose two prime numbers p and q.

Step 2 : Calculate n = p\*q

Step 3 : Calculate  ϕ(n) = (p – 1) \* (q – 1)

Step 4 : Choose e such that gcd(e , ϕ(n) ) = 1 (**1 < e < Φ(n))**

Step 5 : Calculate d such that e\*d mod ϕ(n) = 1

Step 6 : Public Key {e,n} Private Key {d,n}

Step 7 : Cipher text C = Pe mod n  where P = plaintext

Step 8 : For Decryption D = Dd mod n where D will give back the plaintext.

**CODE –**

import java.util.\*;

import java.math.\*;

class harshini\_rsa

{

public static void main(String args[])

{

Scanner sc=new Scanner(System.in);

int p,q,n,z,d=0,e,i;

System.out.println("Enter the number to be encrypted and decrypted");

int msg=sc.nextInt();

double c;

BigInteger msgback;

System.out.println("Enter 1st prime number p");

p=sc.nextInt();

System.out.println("Enter 2nd prime number q");

q=sc.nextInt();

n=p\*q;

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

System.out.println("the value of z = "+z);

for(e=2;e<z;e++)

{

if(gcd(e,z)==1) // e is for public key exponent

{

break;

}

}

System.out.println("the value of e = "+e);

for(i=0;i<=9;i++)

{

int x=1+(i\*z);

if(x%e==0) //d is for private key exponent

{

d=x/e;

break;

}

}

System.out.println("the value of d = "+d);

c=(Math.pow(msg,e))%n;

System.out.println("Encrypted message is : -");

System.out.println(c);

//converting int value of n to BigInteger

BigInteger N = BigInteger.valueOf(n);

//converting float value of c to BigInteger

BigInteger C = BigDecimal.valueOf(c).toBigInteger();

msgback = (C.pow(d)).mod(N);

System.out.println("Derypted message is : -");

System.out.println(msgback);

}

static int gcd(int e, int z)

{

if(e==0)

return z;

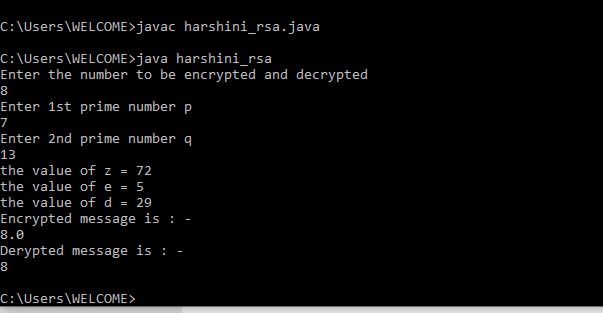
else

return gcd(z%e,e);

}

}

**OUTPUT –**

****

**CONCLUSION :**

Thus RSA algorithm is implemented and studied for cryptographical analysis .

**-HARSHINI T (2017103532)**