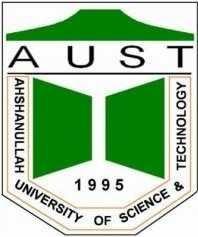
***Ahsanullah University of Science & Technology***

Department of Computer Science & Engineering

**FALL 2020**

****

**Assignment – 01**

**(RSA)**

[**Data Communication**](https://classroom.google.com/c/MzY0NTM0ODY2MTM4)

**CSE 3211**

**Submitted To**

### 

### **Mr. H M Zabir Haque**

**Assistant Professor**

**Department of CSE, AUST**

**Submitted By:**

**Name: Atanu Saha**

**ID: 17.02.04.003**

**Section:** A

**Date of submission:11-09-2021**

package javaapplication6;

import java.math.BigInteger;

import java.util.Random;

import java.util.Scanner;

/\*\*

\*

\* @author Atanu saha

\*/

public class JavaApplication6 {

public static void main(String[] args) {

BigInteger p, q;

Integer bit;

System.out.println("Please Enter Bit Number for p & q: ");

Scanner s = new Scanner(System.in);

bit = s.nextInt();

p = BigInteger.probablePrime(bit, new Random());

q = BigInteger.probablePrime(bit, new Random());

System.out.println("Random Prime Number p: " + p);

System.out.println("Random Prime Number q: " + q);

BigInteger n;

n = p.multiply(q);

System.out.println("n=(p\*q): " + n);

BigInteger one = BigInteger.ONE;

BigInteger psub, qsub;

psub = p.subtract(one);

qsub = q.subtract(one);

BigInteger phi;

phi = psub.multiply(qsub);

BigInteger e1 = BigInteger.TEN;

while (e1.compareTo(phi) < 0) {

if (e1.gcd(phi).equals(BigInteger.ONE)) {

break;

} else {

e1 = e1.add(BigInteger.ONE);

}

}

BigInteger d = e1.modInverse(phi);

//System.out.println("d = e^-1 mod phi: " + d);

BigInteger PP, C, PD;

System.out.println("Please Enter Your Message: ");

s.nextLine();

String message = s.nextLine();

PP = new BigInteger(message.getBytes());

System.out.println("Before Encrypted P: " + PP);

C = PP.modPow(e1, n);

System.out.println("After Encryption C: " + C);

PD = C.modPow(d, n);

System.out.println("After Decryption P: " + PD);

System.out.println("Message After Decryption: " + new String(PD.toByteArray()));

}

}

Output:

