

```

import java.util.Random;

import java.util.Scanner;


public class RSa{

public static void main(String[] args) {

    String pl,cr,dr;

    int x,y,p,q, i = 2, j = 2, n, m, e, d;

    try (Scanner sc = new Scanner(System.in)) {

        Random r = new Random();

        System.out.println("Enter two distances upper limit fo p,q:");

        System.err.println("Enter the first prime number:");

        x = sc.nextInt();

        System.err.println("Enter the second prime number:");

        y = sc.nextInt();

        do {

            p = r.nextInt(x);

        } while (!isPrime(p, i));

        System.out.println("The first prime number (p) is:" + p);

        do {

            q = r.nextInt(y);

        } while (!isPrime(q, j));

        System.out.println("The second prime number (q) is:" + q);

        n = p * q;

        m = (p - 1) * (q - 1);

        System.err.println("The value of (n) is:" + n);

        System.err.println("The value of (m) is:" + m);

        e = r.nextInt(m);

        System.err.println("The value of (e) is:" + e);

        d = modInverse(e, m);
    }
}

```

```

        System.err.println("The value of (d) is:" + d);
        System.err.println("Enter the plain text:");
        pl = sc.next();
    }
    System.err.println("The plain text is:" + pl);
    cr = encrypt(pl, e, n);
    System.err.println("The cipher text is:" + cr);
    dr = decrypt(cr, d, n);
    System.err.println("The decrypted text is:" + dr);
}

```

```

public static boolean isPrime(int n, int i) {
    if (n <= 1) {
        return false;
    } else {
        for (i = 2; i <= n / 2; i++) {
            if (n % i == 0) {
                return false;
            }
        }
        return true;
    }
}

```

```

public static int gcd(int a, int b) {
    int r = 2;
    if (b == 0) {
        return a;
    } else {

```

```

while (r != 0) {
    r = a % b;
    a = b;
    b = r;
}
return a;
}
}

```

```

public static int modInverse(int e, int m) {
    int d;
    for (d = 0; d < m; d++) {
        if ((d * e) % m == 1)
            break;
    }
    return d;
}

```

```

public static String encrypt(String pl, int e, int n) {
    int i, j, k;
    String cr = "";
    char c;
    for (i = 0; i < pl.length(); i++) {
        c = pl.charAt(i);
        j = (int) c;
        k = 1;
        for (int l = 0; l < e; l++) {
            k = k * j;
            k = k % n;

```

```

    }
    cr = cr + (char) k;
}
return cr;
}

```

```

public static String decrypt(String cr, int d, int n) {

```

```

    int i, j, k;
    String dr = "";
    char c;
    for (i = 0; i < cr.length(); i++) {
        c = cr.charAt(i);
        j = (int) c;
        k = 1;
        for (int l = 0; l < d; l++) {
            k = k * j;
            k = k % n;
        }
        dr = dr + (char) k;
    }
    return dr;
}

```

```

public static int modInverse(int message, int e, int n){

```

```

    int pl = 1;
    for (int i = 0; i < e; i++) {

    }
    return pl;
}

```

}