**Lab – 3**

**Subject : NIS**

**Aim:** **Write a Program to do Encryption and Decryption using Vigenere Cipher. Do the Cryptanalysis of Vigenere Cipher (Use sufficiently large CipherText). Use Index of Coincidence to verify the guessed Key Length. Use Mutual Index of Coincidence to guess the Key.**

1. **Write a Program to do Encryption and Decryption using Vigenere Cipher**

**Program: -**

import java.util.\*;

import java.lang.\*;

class VigenereCipher

{

    public static int[] charToInt(String text){

        System.out.println("length : " + text.length());

        int[] convert = new int[text.length()];

        int j=0;

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

            char c = text.charAt(i);

            if(c != ' '){

                int temp =  (int)c - 97;

                convert[j] = temp;

                j++;

              //System.out.println(convert[i]);

            }

        }

      //  System.out.println("count of j : " + j +"\nlength of convert : " + convert.length);

        return convert;

    }

        public static int positiveInvers(int inverse){

            int n=26;

            while(inverse < 0){

                inverse = inverse + n;

            }

            return inverse;

        }

        public static void encryption(String p,String key){

            int[] pInt = charToInt(p);

            int[] keyInt = charToInt(key);

            int enc=0;

            System.out.println("Encryption : ");

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

                enc = ((pInt[i] + keyInt[i % key.length()]) % 26) + 97;

                System.out.print((char)enc);

            }

        }

        public static void decryption(String cipher,String key){

            int[] cipherInt = charToInt(cipher);

            int[] keyInt = charToInt(key);

            int dec=0;

            System.out.println("Decryption : ");

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

                dec = positiveInvers((cipherInt[i] - keyInt[i % key.length()])) + 97;

                System.out.print((char)dec + " ");

            }

        }

        public static void main(String args[])

        {

            Scanner sc = new Scanner(System.in);

            System.out.println("Enter Plain/Cipher text : ");

            String plainText = sc.nextLine();

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

            String key = sc.nextLine();

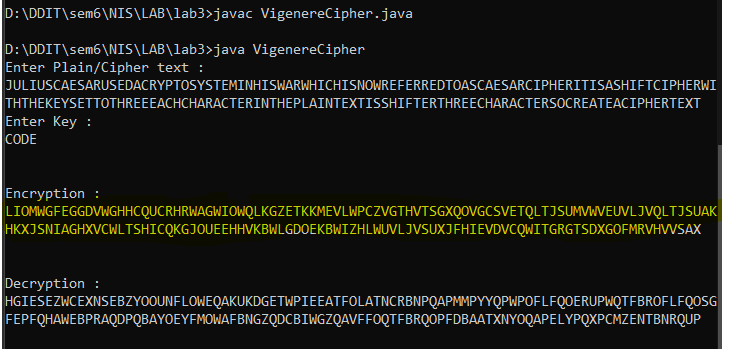
            encryption(plainText,key);

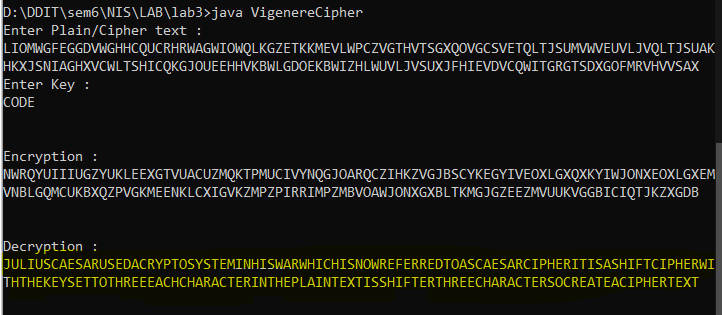
            decryption(plainText,key);

        }

    }

**Output: -**

****

****

1. **Cryptanalysis.**

**Program: -**

import java.util.\*;

import java.lang.\*;

class Cryptanalisys{

    public static void mutulIndexOfCoincidence(char[] Y){

        double english\_freq[]={8.167,1.492,2.782,4.253,12.702,2.228,2.015,6.094,6.996,0.153,0.772,4.025,2.406,6.749,7.507,1.929,0.095,5.987,6.327,9.056,2.758,0.978,2.360,0.150,1.974,0.074};

        double[] p=new double[26];

        double[] q=new double[26];

        double IC = 0.0;

        int[] f = frequency(Y);

        double sum = 0;

        for (int i = 0; i < 26; i++)

        {

            p[i] = english\_freq[i] / 100;

            q[i] =(double)f[i]/Y.length;

           // System.out.println("q[i]" + q[i]);

        }

        int j=0;

        for(int k=0;k<26;k++)

        {

            sum = 0;

            for(int i=0;i<26;i++){

                sum = sum + (p[i] \* q[(i + k) %26 ]);

            }

            System.out.println("Sum : " + sum + "    index : " + (char)((int)j+97));

            j++;

        }

        System.out.println("\n");

    }

    public static int[] frequency(char[] Y){

        int[] f = new int[26];

        for(int i=0;i<Y.length;i++)

        {

            //System.out.print(Y[i]);

            f[(int)(Y[i]-'A')]++;

           // System.out.println("Y["+Y[i]+"] = " + (f[(int)(Y[i]-65)]));

        }

        return f;

    }

    public static void analisys(String ct1){

        int m = 4;

        char[] ct = ct1.toCharArray();

        double size =Math.ceil((double)ct.length/m);

        char[][] Y = new char[m][(int)size];

        double[] IC = new double[m];

        String y1 = "LWGWCRAOKTEPGTQCTJVUEGVGUQGECVPRPVJGTJEUGCJG";

        Y[0]= y1.toCharArray();

        String y2 = "IGGGQHGWGKVCTSOSQSWVWFVYSHSVFSHZHWWFSOHCOQSL";

        Y[1]= y2.toCharArray();

        String y3 = "OFDHURWQZKLZHGVVLUVLSZWHWKHFDUKDHVIWHUHFWLUW";

        Y[2]= y3.toCharArray();

        String y4 = "MEVHCWILEMWVVXGETMEXLMLCXVELGMIMBWXLGEVVITX";

        Y[3]= y4.toCharArray();

        for(int i=0;i<m;i++)

        {

            mutulIndexOfCoincidence(Y[i]);

        }

    }

    public static void main(String args[]){

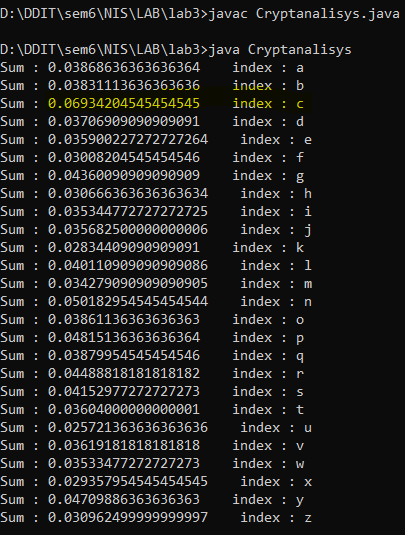
        String ct ="LIOMWGFEGGDVWGHHCQUCRHRWAGWIOWQLKGZETKKMEVLWPCZVGTHVTSGXQOVGCSVETQLTJSUMVWVEUVLXEWSLGFZMVVWLGYHCUSWXQHKVGSHEEVFLCFDGVSUMPHKIRZDMPHHBVWVWJWIXGFWLTSHGJOUEEHHVUCFVGOWICQLTJSUXGLW";

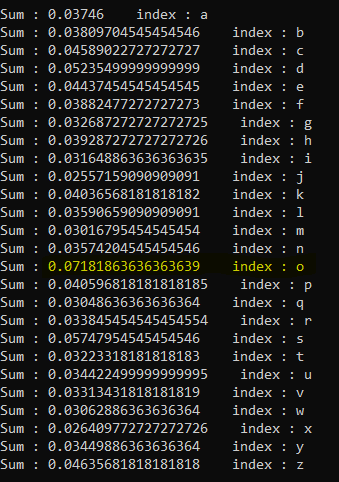
        analisys(ct);

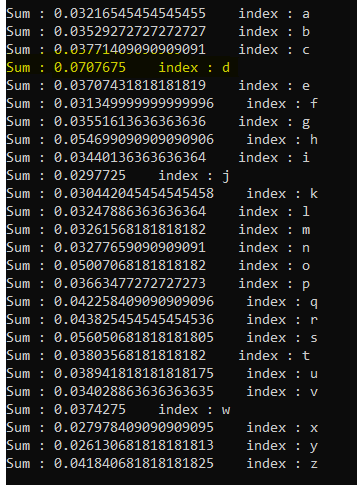
    }

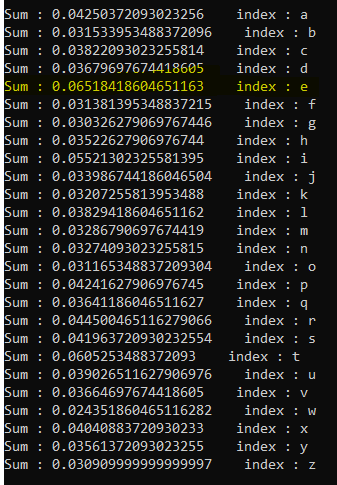
}

**Output: -**

****

****

****

****