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.

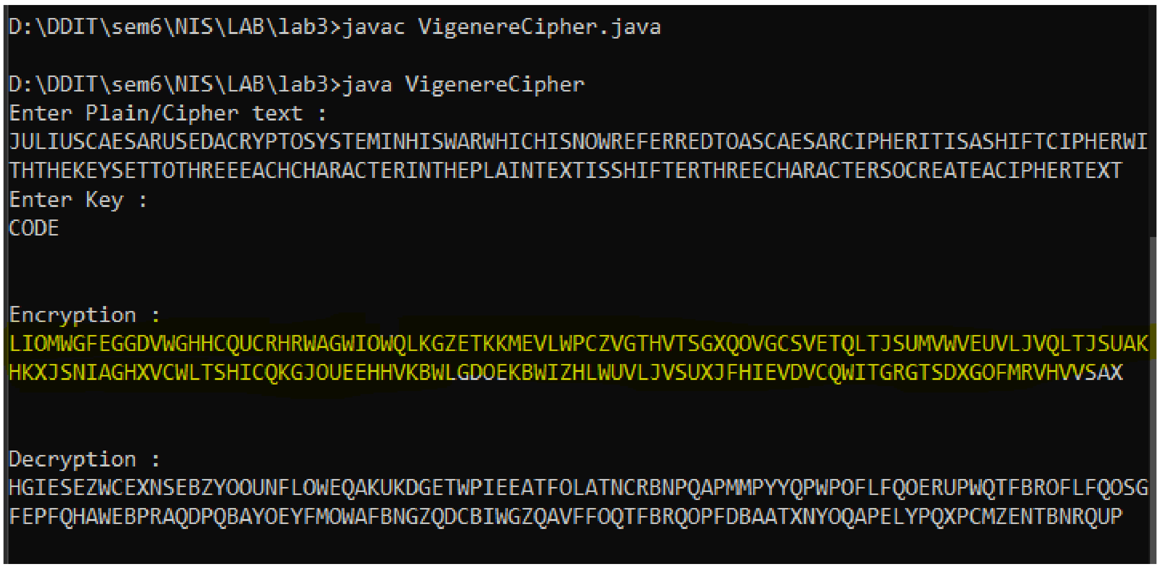
1. **Write a Program to do Encryption and Decryption using Vigenere Cipher**

Program: -

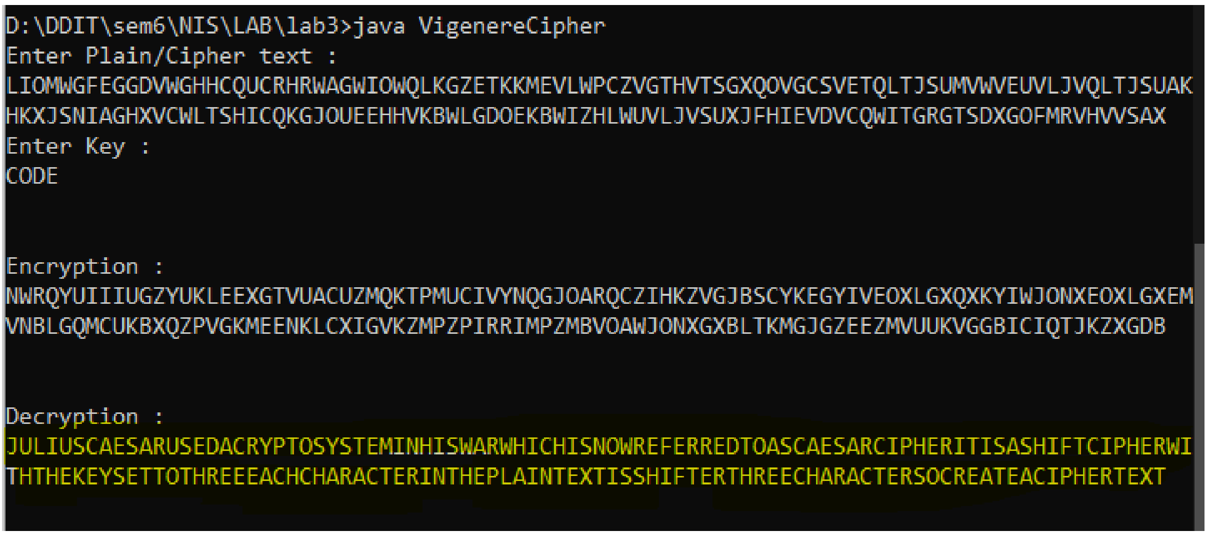
```
import java.util.*;
import java.lang.*;

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);
```

```java
            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: -



```
D:\DDIT\sem6\NIS\LAB\lab3>javac VigenereCipher.java

D:\DDIT\sem6\NIS\LAB\lab3>java VigenereCipher
Enter Plain/Cipher text :
JULIUSCAESARUSEDACRYPTOSYSTEMINHISWARWHICHISNOWREFERREDTOASCAESARCIPHERITISASHIFTCIPHERWI
THTHEKEYSETTOTHREEEACHCHARACTERINTHEPLAINTEXTISSHIFTERTHREECHARACTERSOCREATEACIPHERTEXT
Enter Key :
CODE


Encryption :
LIOMWGFEGGDVWGHHCQUCRHRWAGWIOWQLKGZETKKMEVLWPCZVGTHVTSGXQOVGCSVETQLTJSUMVWVEUVLJVQLTJSUAK
HKXJSNIAGHXVCWLTSHICQKGJOUEEHHVKBWLGDOEKBWIZHLWUVLJVSUXJFHIEVDVCQWITGRGTSDXGOFMRVHVVSAX


Decryption :
HGIESEZWCEXNSEBZYOOUNFLOWEQAKUKDGETWPIEEATFOLATNCRBNPQAPMMPYYQPWPOFLFQOERUPWQTFBROFLFQOSG
FEPFQHAWEBPRAQDPQBAYOEYFMOWAFBNGZQDCBIWGZQAVFFOQTFBRQOPFDBAATXNYOQAPELYPQXPCMZENTBNRQUP
```



```
D:\DDIT\sem6\NIS\LAB\lab3>java VigenereCipher
Enter Plain/Cipher text :
LIOMWGFEGGDVWGHHCQUCRHRWAGWIOWQLKGZETKKMEVLWPCZVGTHVTSGXQOVGCSVETQLTJSUMVWVEUVLJVQLTJSUAK
HKXJSNIAGHXVCWLTSHICQKGJOUEEHHVKBWLGDOEKBWIZHLWUVLJVSUXJFHIEVDVCQWITGRGTSDXGOFMRVHVVSAX
Enter Key :
CODE


Encryption :
NWRQYUIIIUGZYUKLEEXGTVUACUZMQKTPMUCIVYNQGJOARQCZIHKZVGJBSCYKEGYIVEOXLGXQXKYIWJONXEOXLGXEM
VNBLGQMCUKBXQZPVGKMEENKLCXIGVKZMPZPIRRIMPZMBVOAWJONXGXBLTKMGJGZEEZMVUUKVGGBICIQTJKZXGDB


Decryption :
JULIUSCAESARUSEDACRYPTOSYSTEMINHISWARWHICHISNOWREFERREDTOASCAESARCIPHERITISASHIFTCIPHERWI
THTHEKEYSETTOTHREEEACHCHARACTERINTHEPLAINTEXTISSHIFTERTHREECHARACTERSOCREATEACIPHERTEXT
```

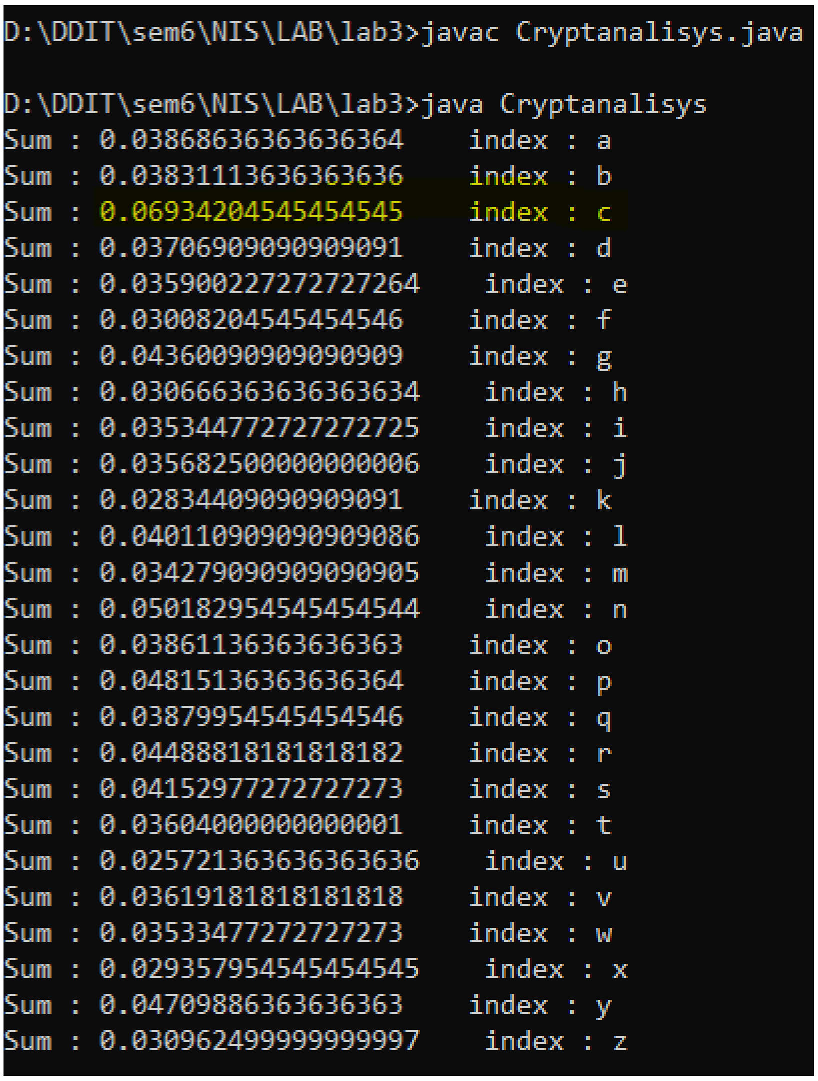
2. **Cryptanalysis.**

Program: -

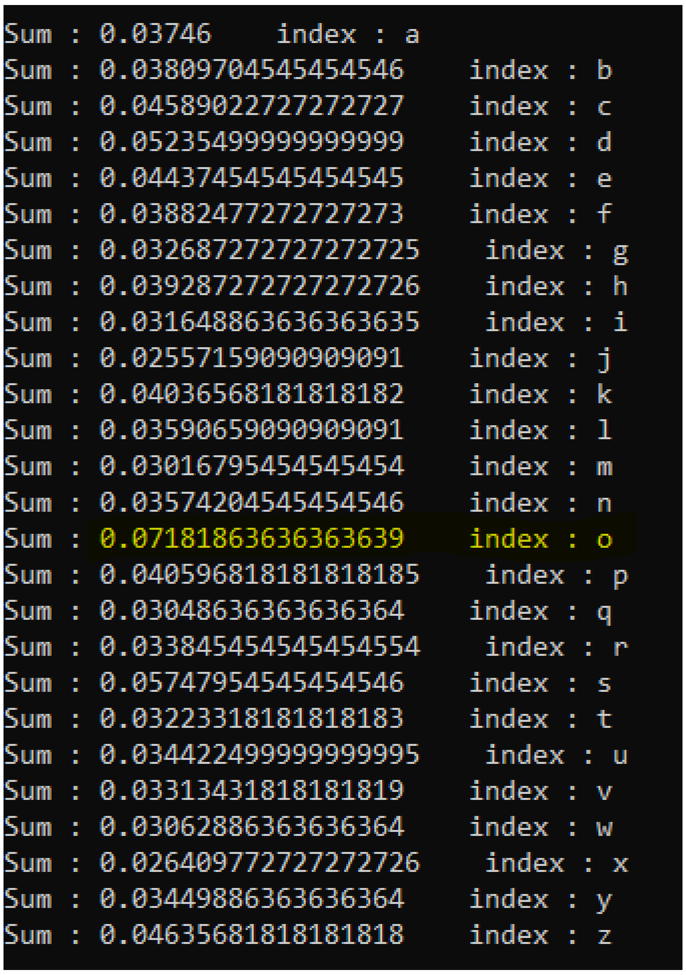
```
import java.util.*;
import java.lang.*;

class Cryptanalisys{

    public static void mutulIndexOfCoincidence(char[] Y){

        double english_freq[]={8.167,1.492,2.782,4.253,12.702,2.228,2.015,6.09
4,6.996,0.153,0.772,4.025,2.406,6.749,7.507,1.929,0.095,5.987,6.327,9.056,2.75
8,0.978,2.360,0.150,1.974,0.074};
        double[] p=new double[26];
        double[] q=new double[26];
```

```java
        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 ="LIOMWGFEGGDVWGHHCQUCRHRWAGWIOWQLKGZETKKMEVLWPCZVGTHVTSGXQO
VGCSVETQLTJSUMVWVEUVLXEWSLGFZMVVWLGYHCUSWXQHKVGSHEEVFLCFDGVSUMPHKIRZDMPHHBVWVW
JWIXGFWLTSHGJOUEEHHVUCFVGOWICQLTJSUXGLW";
        analisys(ct);
    }

}
```

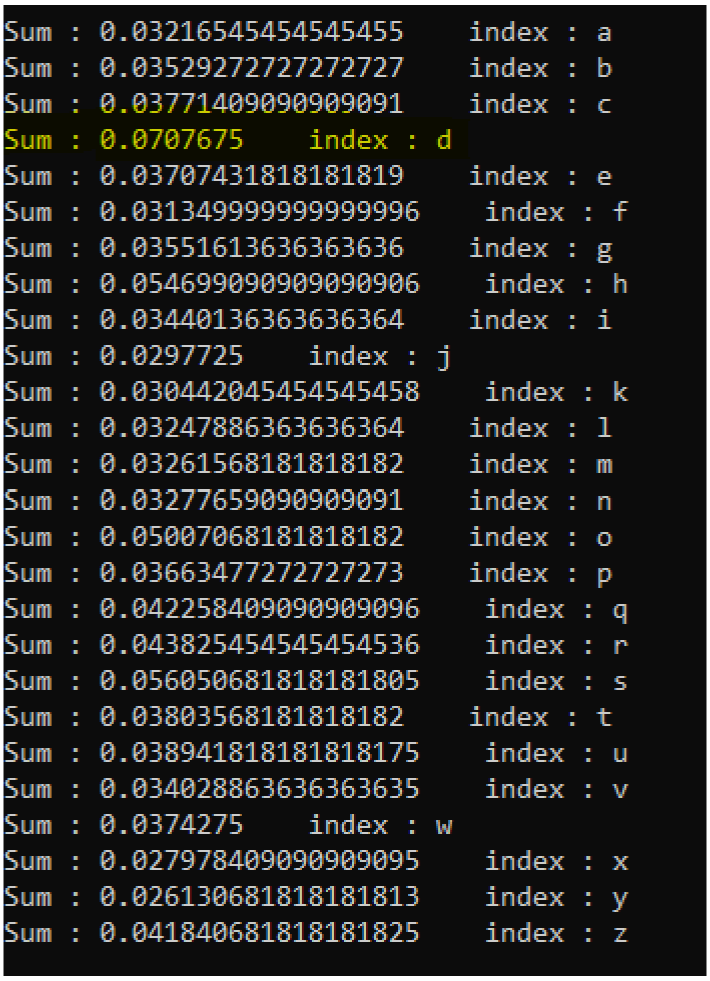
Output: -



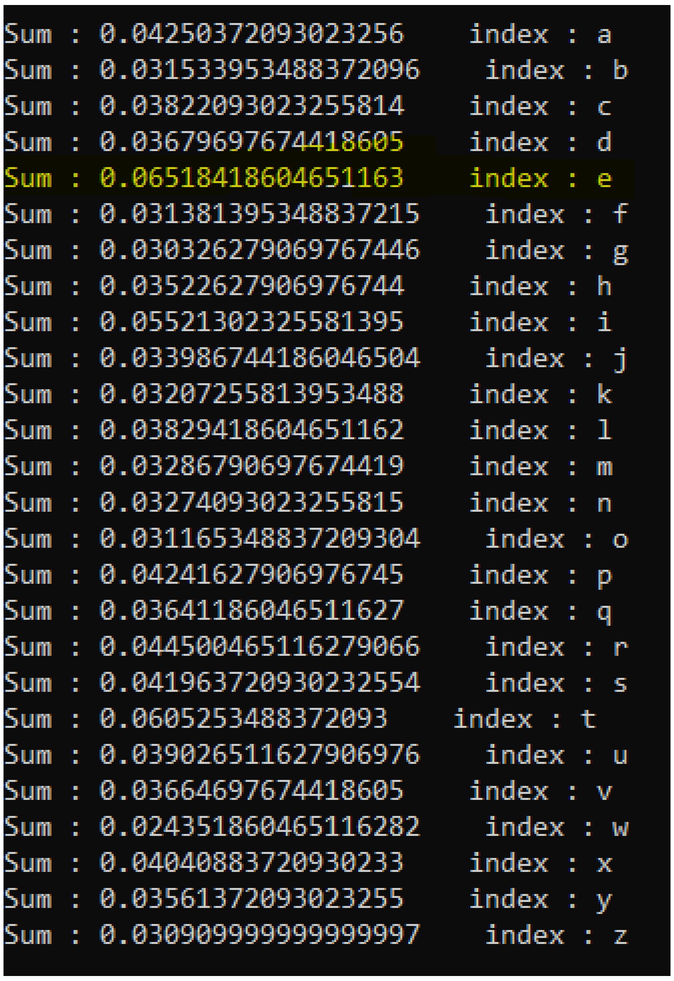
```
D:\DDIT\sem6\NIS\LAB\lab3>javac Cryptanalisys.java

D:\DDIT\sem6\NIS\LAB\lab3>java Cryptanalisys
Sum : 0.038686363636363 64    index : a
Sum : 0.0383111363636363 6    index : b
Sum : 0.0693420454545454 5    index : c
Sum : 0.0370690909090909 1    index : d
Sum : 0.03590022727272727264    index : e
Sum : 0.0300820454545454 6    index : f
Sum : 0.0436009090909090 9    index : g
Sum : 0.030666363636363634    index : h
Sum : 0.035344772727272725    index : i
Sum : 0.035682500000000006    index : j
Sum : 0.0283440909090909 1    index : k
Sum : 0.040110909090909086    index : l
Sum : 0.034279090909090905    index : m
Sum : 0.050182954545454544    index : n
Sum : 0.0386113636363636 3    index : o
Sum : 0.0481513636363636 4    index : p
Sum : 0.0387995454545454 6    index : q
Sum : 0.0448881818181818 2    index : r
Sum : 0.0415297727272727 3    index : s
Sum : 0.0360400000000000 1    index : t
Sum : 0.025721363636363636    index : u
Sum : 0.0361918181818181 8    index : v
Sum : 0.0353347727272727 3    index : w
Sum : 0.029357954545454545    index : x
Sum : 0.0470988636363636 3    index : y
Sum : 0.030962499999999997    index : z
```



```
Sum : 0.03746    index : a
Sum : 0.03809704545454546    index : b
Sum : 0.04589022727272727    index : c
Sum : 0.05235499999999999    index : d
Sum : 0.04437454545454545    index : e
Sum : 0.03882477272727273    index : f
Sum : 0.032687272727272725    index : g
Sum : 0.039287272727272726    index : h
Sum : 0.031648863636363635    index : i
Sum : 0.02557159090909091    index : j
Sum : 0.04036568181818182    index : k
Sum : 0.03590659090909091    index : l
Sum : 0.03016795454545454    index : m
Sum : 0.03574204545454546    index : n
Sum : 0.07181863636363639    index : o
Sum : 0.040596818181818185    index : p
Sum : 0.03048636363636364    index : q
Sum : 0.033845454545454554    index : r
Sum : 0.05747954545454546    index : s
Sum : 0.03223318181818183    index : t
Sum : 0.034422499999999995    index : u
Sum : 0.03313431818181819    index : v
Sum : 0.03062886363636364    index : w
Sum : 0.026409772727272726    index : x
Sum : 0.03449886363636364    index : y
Sum : 0.04635681818181818    index : z
```



```
Sum : 0.03216545454545455     index : a
Sum : 0.03529272727272727     index : b
Sum : 0.03771409090909091     index : c
Sum : 0.0707675     index : d
Sum : 0.03707431818181819     index : e
Sum : 0.031349999999999996     index : f
Sum : 0.03551613636363636     index : g
Sum : 0.054699090909090906     index : h
Sum : 0.03440136363636364     index : i
Sum : 0.0297725     index : j
Sum : 0.030442045454545458     index : k
Sum : 0.03247886363636364     index : l
Sum : 0.03261568181818182     index : m
Sum : 0.03277659090909091     index : n
Sum : 0.05007068181818182     index : o
Sum : 0.03663477272727273     index : p
Sum : 0.042258409090909096     index : q
Sum : 0.043825454545454536     index : r
Sum : 0.056050681818181805     index : s
Sum : 0.03803568181818182     index : t
Sum : 0.038941818181818175     index : u
Sum : 0.034028863636363635     index : v
Sum : 0.0374275     index : w
Sum : 0.027978409090909095     index : x
Sum : 0.026130681818181813     index : y
Sum : 0.041840681818181825     index : z
```



```
Sum : 0.04250372093023256    index : a
Sum : 0.031533953488372096    index : b
Sum : 0.03822093023255814    index : c
Sum : 0.03679697674418605    index : d
Sum : 0.06518418604651163    index : e
Sum : 0.031381395348837215    index : f
Sum : 0.030326279069767446    index : g
Sum : 0.03522627906976744    index : h
Sum : 0.05521302325581395    index : i
Sum : 0.033986744186046504    index : j
Sum : 0.03207255813953488    index : k
Sum : 0.03829418604651162    index : l
Sum : 0.03286790697674419    index : m
Sum : 0.03274093023255815    index : n
Sum : 0.031165348837209304    index : o
Sum : 0.04241627906976745    index : p
Sum : 0.03641186046511627    index : q
Sum : 0.044500465116279066    index : r
Sum : 0.041963720930232554    index : s
Sum : 0.0605253488372093    index : t
Sum : 0.039026511627906976    index : u
Sum : 0.03664697674418605    index : v
Sum : 0.024351860465116282    index : w
Sum : 0.04040883720930233    index : x
Sum : 0.03561372093023255    index : y
Sum : 0.030909999999999997    index : z
```