

Lab 3

- **Program**
 - **Write a Program to do Encryption and Decryption using Vigenere Cipher. 2. Do the Cryptanalysis of Vigenere Cipher (Use sufficiently large Cipher Text). Use Index of Coincidence to verify the guessed Key Length. Use Mutual Index of Coincidence to guess the Key.**
 - **Code:**

```
import java.util.Arrays;

public class VigenereCipher
{
    public String generateNewKey(String plainText, String key)
    {
        char[] newKey = new char[plainText.length()];

        for(int i = 0, j = 0; i < plainText.length(); i += 1, j += 1)
        {
            if(j == key.length())
                j = 0;
            newKey[i] = key.charAt(j);
        }
        return String.valueOf(newKey);
    }

    public String encryption(String plainText, String newKey)
    {
        char[] cipherText = new char[plainText.length()];

        for(int i = 0; i < plainText.length(); i++)
        {
            cipherText[i] = (char)((((int)plainText.charAt(i) +
            (int)newKey.charAt(i)) % 26) + 'A');
        }
        return String.valueOf(cipherText);
    }
}
```

```

public String decryption(String cipherText,String newKey)
{
    char[] plainText = new char[cipherText.length()];

    for(int i = 0; i < cipherText.length(); i++)
    {
        plainText[i] = (char)((((int)cipherText.charAt(i) -
(int)newKey.charAt(i) + 26) % 26) + 'A');
    }

    return String.valueOf(plainText);
}

public void cryptanalysis()
{
    double
english_freq[]={8.167,1.492,2.782,4.253,12.702,2.228,2.015,6.094,6.996,0.
153,0.772,4.025,2.406,6.749,7.507,1.929,0.095,5.987,6.327,9.056,2.758,0.9
78,2.360,0.150,1.974,0.074};
    double[] p = new double[26];

    for(int i = 0; i < 26; i++)
    {
        p[i] = english_freq[i] / 100;
    }

    //char
Y1[]={ 'C','T','L','S','S','Y','N','Z','T','R','T','Y','M','O','R','T','T','I','N','E','O','E','T'
' };
    //char
Y1[]={ 'R','A','Y','T','T','O','A','T','N','M','O','S','S','R','T','U','H','D','A','C','F','S',
'E' };
    // char
Y1[]={ 'Y','N','S','H','U','F','L','N','F','A','N','T','I','D','O','D','E','D','S','T','T','Y','
M' };
    char
Y1[]={ 'P','A','T','E','D','A','Y','G','O','T','S','E','N','E','S','Y','H','E','P','S','H','S','
S' };

    int[] freq = new int[26];

```

```

Arrays.fill(freq, 0);

double[] q = new double[26];

for(int i = 0; i < Y1.length; i++)
{
    freq[Y1[i] - 65] = freq[Y1[i] - 65] + 1;
}

for(int i = 0; i < 26; i++)
{
    q[i] = (double)(freq[i]) / Y1.length;
}

int a = 65;
for(int k = 0; k < 26; k++)
{
    double sum = 0.0;

    for(int i = 0; i < 26; i++)
    {
        sum = sum + (p[i] * q[(i + k) % 26]);
    }
    System.out.println("Sum = "+sum+" and k = "+k+"["+(char)a+"]");
    a += 1;
}

public static void main(String[] args)
{
    VigenereCipher vc = new VigenereCipher();

    String plainText =
"CRYPTANALYSISISTHESTUDYOFANALYZINGINFORMATIONSYST
EMSINORDERTOSTUDYTHEHIDDENASPECTSOFTHESYSTEMS";

    String key = "LION";

    System.out.println("Your plain texts are: "+plainText);

    String newKey = vc.generateNewKey(plainText, key);

```

Output 1:

Cryptanalysis for Group 1

- **Output 2:**

Cryptanalysis for Group 1

- **Output 3:**

Cryptanalysis of Grop 3

- **Output4:**

Cryptanalysis of Grop 4