14.08.2020

Ex No: 1.C

IMPLEMENTATION OF HILL CIPHER

AIM:

To write a C program to implement the hill cipher substitution techniques.

DESCRIPTION:

Each letter is represented by a number modulo 26. Often the simple scheme A = 0, B

= 1... Z = 25, is used, but this is not an essential feature of the cipher. To encrypt a message, each block of *n* letters is multiplied by an invertible *n* × *n*matrix, against modulus 26. To decrypt the message, each block is multiplied by the inverse of the matrix used for encryption. The matrix used for encryption is the cipher key, and it should be chosen randomly from the set of invertible *n* × *n* matrices (modulo 26).

ALGORITHM:

STEP-1: Read the plain text and key from the user.

STEP-2: Split the plain text into groups of length three.

STEP-3: Arrange the keyword in a 3*3 matrix.

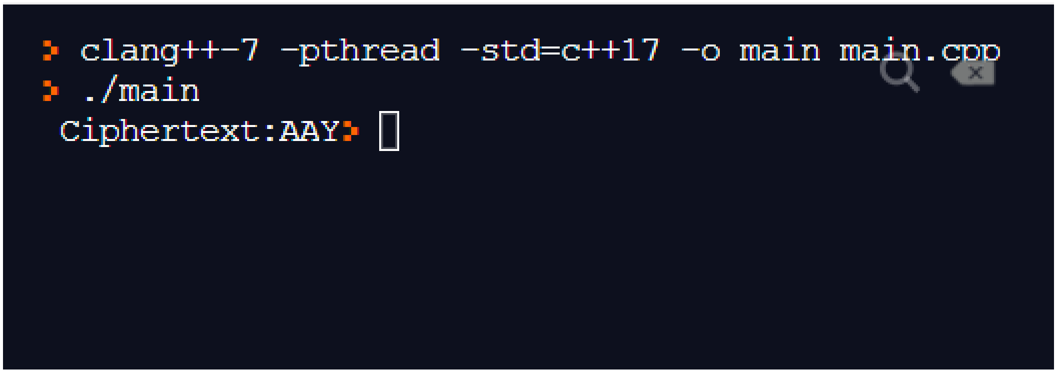
STEP-4: Multiply the two matrices to obtain the cipher text of length three.

STEP-5: Combine all these groups to get the complete cipher text.

PROGRAM: (Caesar Cipher)

```
#include <iostream>
using namespace std;

// Following function generates the
//  key matrix for the key string
void getKeyMatrix(string key, int keyMatrix[][3])
{
    int k = 0;
    for (int i = 0; i < 3; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            keyMatrix[i][j] = (key[k]) % 65;
            k++;
        }
    }
}

// Following function encrypts the message
void encrypt(int cipherMatrix[][1],
             int keyMatrix[][3],
             int messageVector[][1])
{
    int x, i, j;
    for (i = 0; i < 3; i++)
    {
        for (j = 0; j < 1; j++)
        {
            cipherMatrix[i][j] = 0;

             for (x = 0; x < 3; x++)
            {
                cipherMatrix[i][j] +=
                     keyMatrix[i][x] * messageVector[x][j];
            }

            cipherMatrix[i][j] = cipherMatrix[i][j] % 26;
        }
    }
}

// Function to implement Hill Cipher
void HillCipher(string message, string key)
{
    // Get key matrix from the key string
    int keyMatrix[3][3];
    getKeyMatrix(key, keyMatrix);

    int messageVector[3][1];

    // Generate vector for the message
    for (int i = 0; i < 3; i++)
        messageVector[i][0] = (message[i]) % 65;

    int cipherMatrix[3][1];

    // Following function generates
    // the encrypted vector
    encrypt(cipherMatrix, keyMatrix, messageVector);

    string CipherText;

    // Generate the encrypted text from
    // the encrypted vector
    for (int i = 0; i < 3; i++)
        CipherText += cipherMatrix[i][0] + 65;

    // Finally print the ciphertext
    cout << " Ciphertext:" << CipherText;
}

// Driver function for above code
int main()
{
    // Get the message to be encrypted
    string message = "Geedha.India";

    // Get the key
    string key = "GYBNQKURP";

    HillCipher(message, key);

    return 0;
}
```

OUTPUT:



```
› clang++-7 -pthread -std=c++17 -o main main.cpp
› ./main
 Ciphertext:AAY›
```

RESULT:

Thus the hill cipher substitution technique had been implemented successfully in C.