**4CS372 : Cryptography and Network Security**

**B.Tech. (CSE) – I [ 2022-23 ]**

**Assignment No - 4**

**Implementation of Vigenere Cipher**

**Aim: Implementation of Vigenere Cipher in cpp**

**Theory: Vigenere Cipher is type of Substituation Cipher.In which there is 1 key and 1 plain text.Where key is Multiplied as length of plain text. And then index value is calculated of both plain and key and addition is performed at the end**

**Example :**

**Plain text -> Vaishnavi**

**Key -> Vaish**

**Then remaining letter duplicated from start in key.**

**vaish**vais

**vaishnavi**

**Code:**

#include<bits/stdc++.h>

using namespace std;

class Vigenere

{

    public:

        string key;

    void createkey(string k) {

        key.clear();

        for (int i = 0; i < k.size(); ++i)

        {

            if (k[i] >= 'A' && k[i] <= 'Z')

                key += k[i];

            else if (k[i] >= 'a' && k[i] <= 'z')

                key += k[i] + 'A' - 'a';

        }

    }

    string encryption(string t)

    {

        string output;

        for (int i = 0, j = 0; i < t.length(); ++i)

        {

            char c = t[i];

            if(c == ' ')

                continue;

            if (c >= 'a' && c <= 'z')

                c += 'A' - 'a';

            else if (c < 'A' || c > 'Z')

                continue;

            output += (c + key[j] - 2 \* 'A') % 26 + 'A';

            j = (j + 1) % key.length();

        }

        return output;

    }

    string decryption(string t)

    {

        string output;

        for (int i = 0, j = 0; i < t.length(); ++i)

        {

            char c = t[i];

            if (c >= 'a' && c <= 'z')

                c += 'A' - 'a';

            else if (c < 'A' || c > 'Z')

                continue;

            output += (c - key[j] + 26) % 26 + 'A';

            j = (j + 1) % key.length();

        }

        return output;

    }

};

int main()

{

    Vigenere v;

    int choice;

    int datachoice;

    string sample,key;

    int shift;

    while(1)

    {

        cout << "Vigenere Cipher\n 1. Encryption \n 2. Decryption\n 3. Exit\nEnter Choice: ";

        cin>>choice;

        if(choice>2)

            break;

        switch(choice)

        {

            case 1:

                cout << "Data is from\n 1. Manual Entering \n 2. File \nEnter Choice: ";

                cin>>datachoice;

                if(datachoice == 1)

                {

                    cout<<"Enter data to be Encrypted:\n";

                    cin.ignore();

                    getline(cin,sample);

                    cout<<"Enter the key: ";

                    getline(cin,key);

                    v.createkey(key);

                    cout<<"Encrypted String:\n";

                    cout<<v.encryption(sample)<<endl;

                }

                else

                {

                    cout<<"Enter File Name:\n";

                    cin.ignore();

                    getline(cin,sample);

                    cout<<"Enter the key: ";

                    getline(cin,key);

                    v.createkey(key);

                    fstream myfile;

                    myfile.open(sample.c\_str());

                    string str,s;

                    if(!myfile.is\_open())

                        cout << "Error while Opening File";

                    while(getline(myfile,str))

                        s+=str;

                    myfile.close();

                    s=v.encryption(s);

                    myfile.open("CipherText.txt",ios\_base::out);

                    if(myfile.is\_open())

                        myfile.write(s.data(),s.size());

                    cout<<"File Encrypted\n";

                    myfile.close();

                }

                break;

            case 2:

                cout << "Data is from\n 1. Manual Entering \n 2. File \nEnter Choice: ";

                cin>>datachoice;

                if(datachoice == 1)

                {

                    cout<<"Enter data to be Decrypted:\n";

                    cin.ignore();

                    getline(cin,sample);

                    cout<<"Enter the key: ";

                    getline(cin,key);

                    v.createkey(key);

                    cout<<"Decrypted String:\n";

                    cout<<v.decryption(sample)<<endl;;

                }

                else

                {

                    cout<<"Enter File Name:\n";

                    cin.ignore();

                    getline(cin,sample);

                    cout<<"Enter the key: ";

                    getline(cin,key);

                    v.createkey(key);

                    fstream myfile;

                    myfile.open(sample.c\_str());

                    string str,s;

                    if(!myfile.is\_open())

                        cout << "Error while Opening File";

                    while(getline(myfile,str))

                        s+=str;

                    myfile.close();

                    s=v.decryption(s);

                    myfile.open("PlainText.txt",ios\_base::out);

                    if(myfile.is\_open())

                        myfile.write(s.data(),s.size());

                    cout<<"File Decrypted\n";

                    myfile.close();

                }

                break;

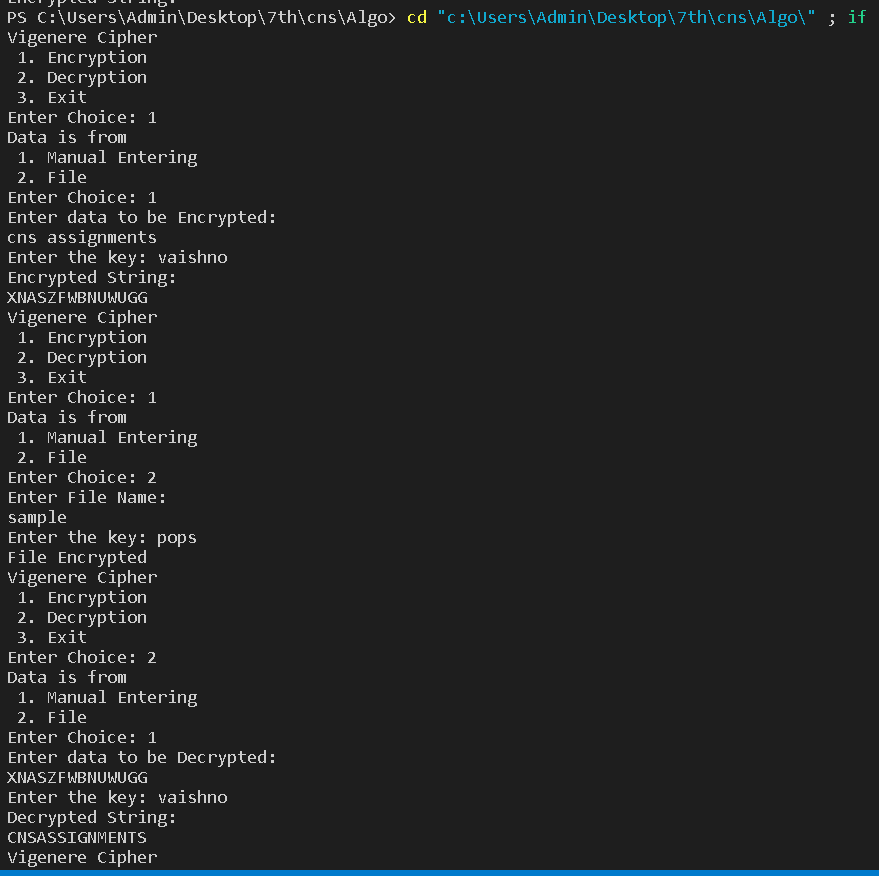
        }

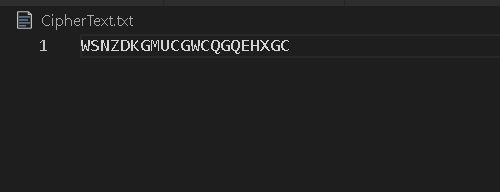
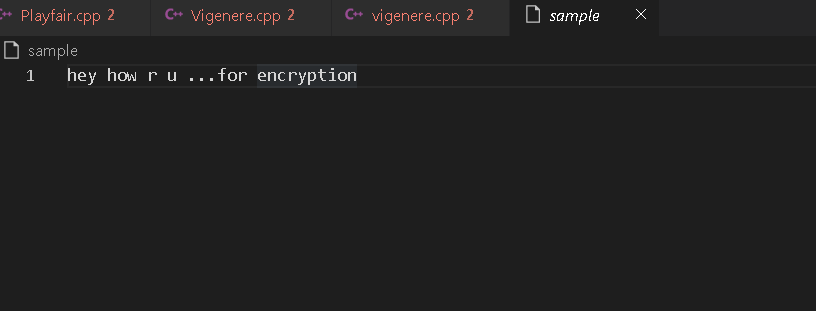
    }

    return 0;

}

**Output:**





**Conclusion:**

**So this cipher not so easy to crack as compare to caesar and not so complicated.**