Instituto Politécnico Nacional
Escuela Superior de Cómputo
Cryptography

Practice 3: Vigenere Cipher

By:
Moreno Zárate Víctor Gibrán

Professor:
M. en C. NIDIA ASUNCIÓN CORTEZ DUARTE
September 2016

Contents

Problem

What is the behavior of the Vigenere cipher?

What are the main characteristics of the Vigenere cipher?

How does the output behave according to the input and the key?

Vigenère cipher is a method of encrypting alphabetic text by using a series of different Caesar ciphers based on the letters of a keyword. It is a simple form of polyalphabetic substitution. [1]

In these method, we use a word as a key, and we add the numerical value of each letter of the message to the numerical value of each letter of the key in order to encrypt. When we want to decrypt we add the adding inverse of each letter.

Hypothesis

A possible solution to the problem is to write a computer program in order to analyze the behavior of the Vigenere cipher, this program will be able to encrypt and decrypt a text using this technique. It also will fix the key if its length is less than the length of the message.

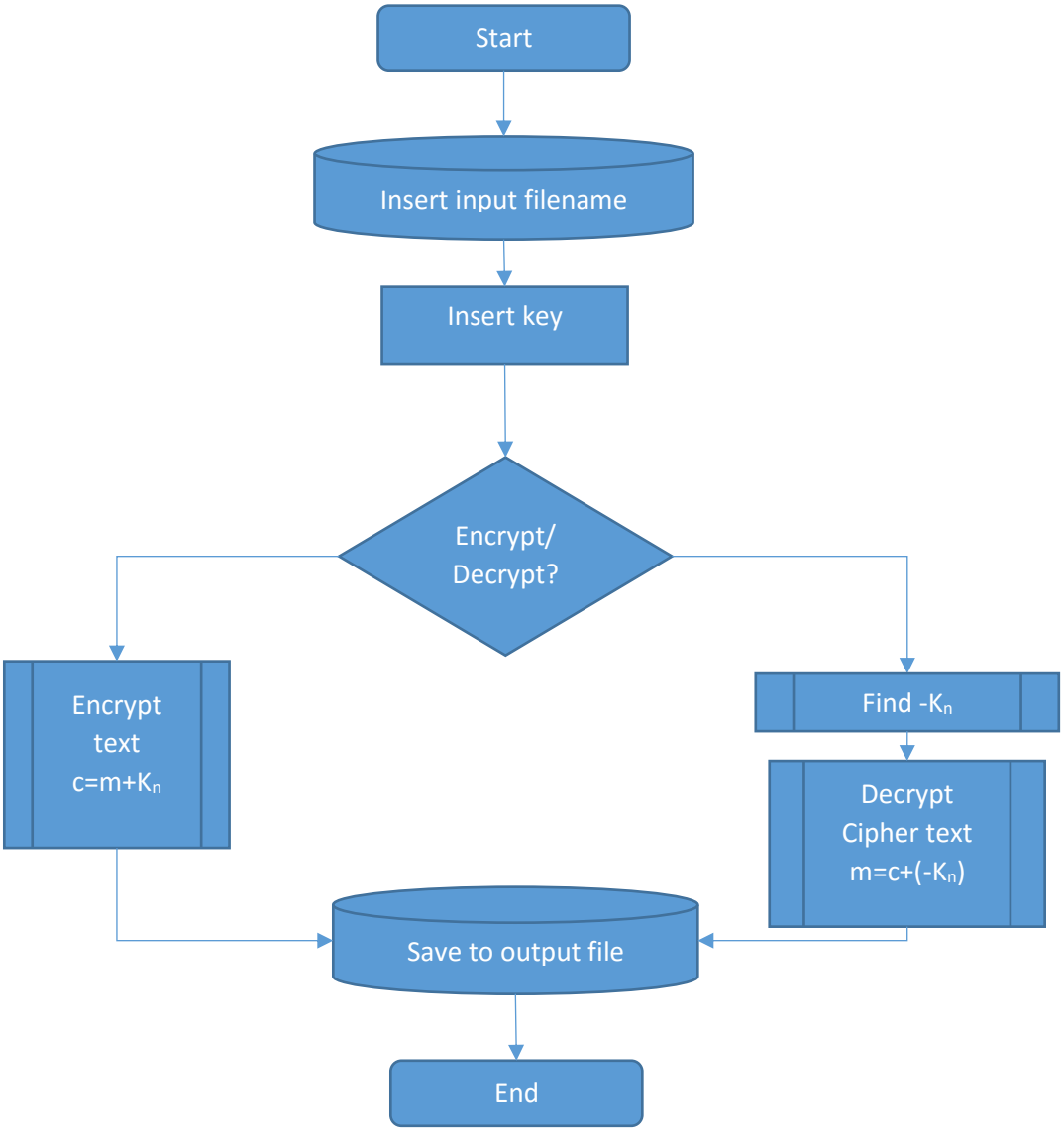
If we program this software correctly, we will see how the character data changes according to the key. Then we will observe the encrypt/decrypt of the cipher text if we put the correct key.

Software (libraries, package, tools)

In order to do this practice, we used:

- Personal Computer
- Linux Operating System
- Text Editor
- GNU C++ Compiler

Procedure



Results (Data)

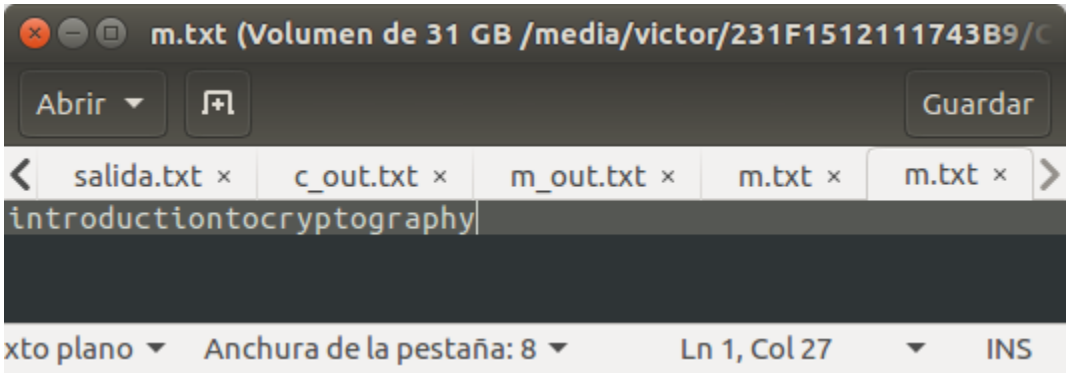


Figure 1: Original text

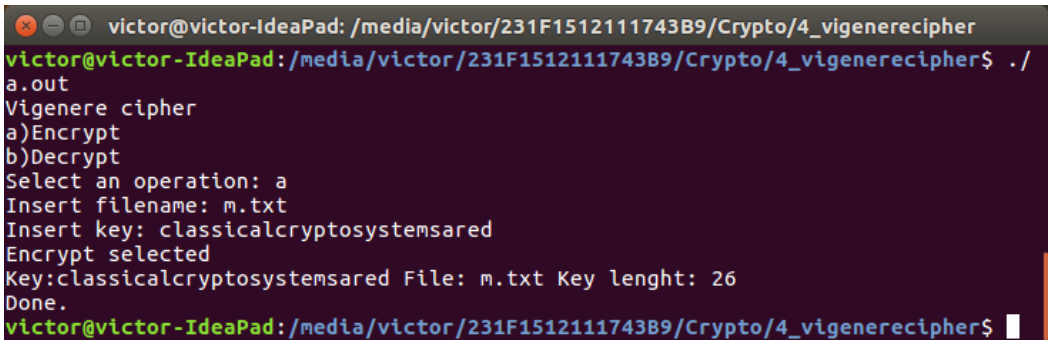


Figure 2: Encryption process

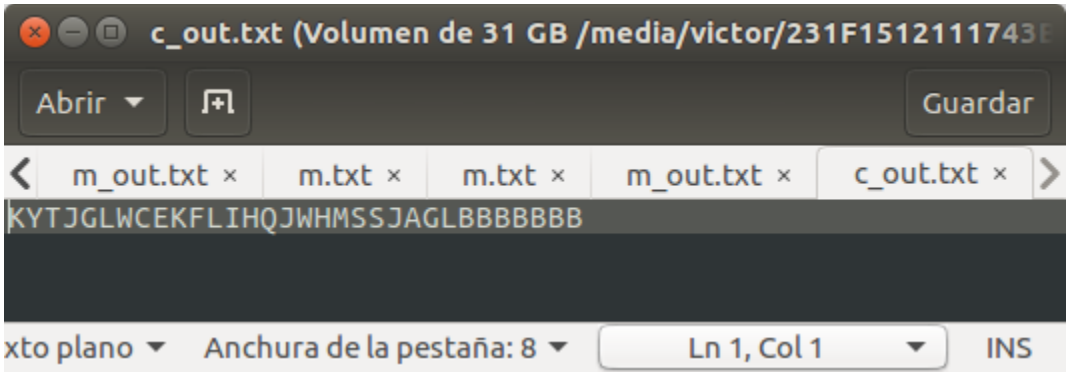


Figure 3: Encrypted data with extra characters at the end of the string.

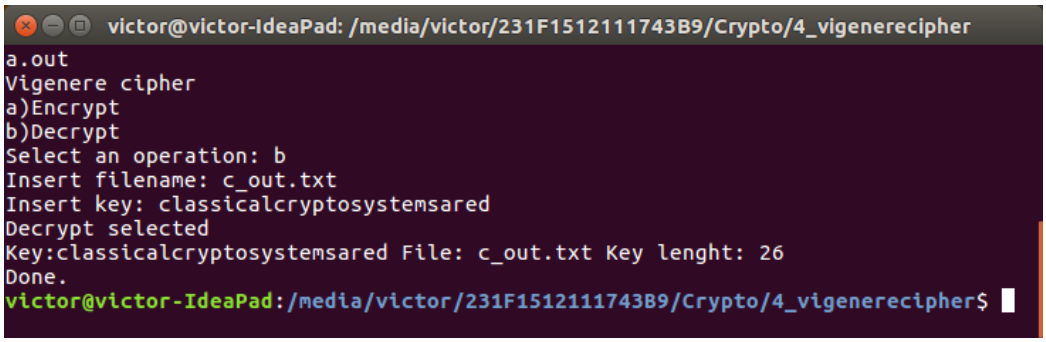


Figure 4: Decrypt process

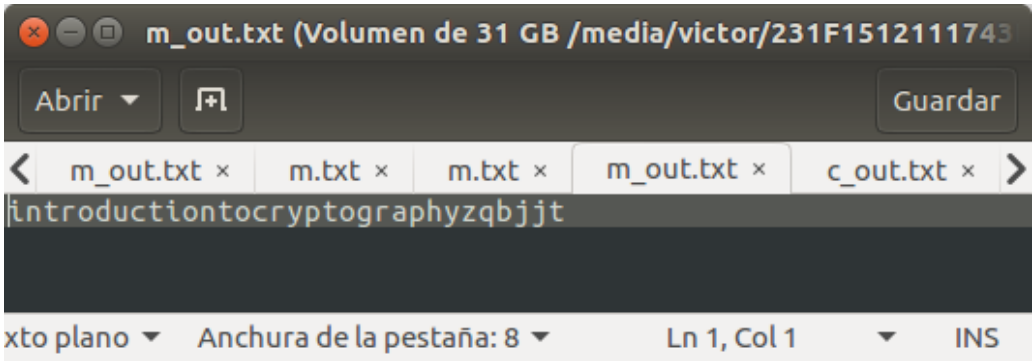


Figure 5: Resultant text.

Conclusions

In this practice, I accepted the hypothesis because I was able to observe the behavior of the Vigenere cipher, and in the program I was able to encrypt and decrypt the message with the correct key, also if we put extra characters of the same letter at the end of the cipher text, it starts to produce different output characters, the reason behind this is because Vigenere cipher is a polyalphabetic method.

I learn the use and implementation of this basic cryptographic method, where I used basic operations with characters of plain text performed by a computer program, and it was an introduction to more complex methods. It can be used in real life to encrypt/decrypt simple plain texts.

Reference

[1] “Vigenère cipher", *Wikipedia*, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Vigen%C3%A8re_cipher . [Accessed: 13- Sep- 2016].

Code

```
#include <iostream>
#include <fstream>
#include <cctype>

using namespace std;

void encrypt (ifstream &input_file,string key,char *letters_uppercase);
void decrypt (ifstream &input_file,string key,char *letters_lowercase);
int addInverse(int n);

int main(){
    char letters_lowercase[]={'a','b','c','d','e','f','g','h','i','j','k','l','m','n','o','p','q','r','s','t','u','v','w','x','y','z'};
    char letters_uppercase[]={'A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z'};
    //Menu
    cout<<"Vigenere cipher\n";
    cout<<"a)Encrypt\n";
    cout<<"b)Decrypt\n";
    cout<<"Select an operation: ";
    char opt;
    cin>>opt;
    cout<<"Insert filename: ";
    char filename[256];
    cin>>filename;
    cout<<"Insert key: ";
    string key;
    cin>>key;
    //Opening input file
    ifstream input_file;
    input_file.open(filename,ios::in|ios::ate);

    if(opt=='a'){
        cout<<"Encrypt selected\n";
```

```
        cout<<"Key:"<<key<<" File: "<<filename<<" Key lenght: "<<key.length()<<end
l;
        encrypt(input_file,key,letters_uppercase);
        cout<<"Done."<<endl;
    }else if(opt=='b'){
        cout<<"Decrypt selected\n";
        cout<<"Key:"<<key<<" File: "<<filename<<" Key lenght: "<<key.length()<<end
l;
        decrypt(input_file,key,letters_lowercase);
        cout<<"Done."<<endl;

    }else{
        cout<<"Invalid option";
    }
    return 0;
}


void encrypt(ifstream &input_file,string key,char *letters_uppercase){
    //Creating output file
    ofstream output_file;
    output_file.open ("c_out.txt",ios::out);
    //Buffer
    int filesize;
    filesize = input_file.tellg();
    char *buffer= new char [filesize];
    input_file.seekg (0, ios::beg);
    input_file.read (buffer, filesize);
    input_file.close();
    //Encrypt
    for(int i=0;i<filesize;i++){
        if(islower(buffer[i])){
        output_file<<letters_uppercase[((buffer[i]-97) + (key[i%key.length()]-
97))%26];
        }
    }
    //Closing stream
    output_file.close();
    delete buffer;
    return;
    }

void decrypt (ifstream &input_file,string key,char *letters_lowercase){
    //Creating output file
    ofstream output_file;
    output_file.open ("m_out.txt",ios::out);
    //Buffer
    int filesize;
    filesize = input_file.tellg();
    char *buffer= new char [filesize];
    input_file.seekg (0, ios::beg);
    input_file.read (buffer, filesize);
    input_file.close();
    //Decrypt
    for(int i=0;i<filesize;i++){
        if(isupper(buffer[i])){
        output_file<<letters_lowercase[((buffer[i]-
65) + addInverse(key[i%key.length()]-97))%26];
        }
    }
    //Closing stream
```

```
90.     output_file.close();
91.     delete buffer;
92.     return;
93.     }
94.
95. int addInverse(int n){
96.     return 26-n;
97.
98.     }
```

Seal of approval