C&NS Lab Assignment 4

Onkar Santosh Gavali (2019BTECS00037)

Batch B2

Index

Rail fence cipher

It is a transposition cipher. (here the position of letters was changed). The plain text is written down as a sequence of diagonals and then read off as a sequence of rows.

Ex “i am the best” depth =2

I		m		h		b		s	
	a		t		e		e		t

Cipher text: IMHBSATEET

Code

railfence.cpp railFence.cpp

railFence.cpp

```cpp
// Cryptography and Network Security Lab
// Assignment 4
// Onkar Gavali
// 2019BTECS00037
// Batch B2
// The program implements the railfence cipher algorithm


#include <iostream>
#include <iomanip>
#include<bits/stdc++.h>
using namespace std;

char upper(char c){
    if(c>='a' && c<='z') return 'A'+c-'a';
}
string encrypt(string plainText, size_t depth){
    string encryption="";
    map<int,string> m;
    int flag=+1,d=0;
    for(int i=0;i<plainText.length();++i){
        if(plainText[i]==' ')continue;
        m[d]+=upper(plainText[i]);
        if(d==depth-1){
            flag=-1;
```

```
        }else if(d==0){
            flag=1;
        }
        d+=flag;
    }
    for(auto a:m){
        encryption+=a.second;
    }
    /*for(size_t i = 0; i < depth; i++){
        size_t position = i;
        for(; position < plainText.size(); position+=depth){
            encryption += upper(plainText[position]);
        }
    }*/
    return encryption;


}

string decrypt(string encryptedText, size_t depth){
    string decryption{};

    size_t jumpSize{0};
    if(encryptedText.size() % depth == 0){
        jumpSize = encryptedText.size()/depth;
```

```
    }else{
        jumpSize = encryptedText.size()/depth + 1;
    }
    int temp = (encryptedText.size() % depth);
    for(size_t i = 0; i < jumpSize; i++){
        size_t pos = i;
        int currStart = i;
        int count = 1;
        for(; pos < encryptedText.size();){
            decryption += encryptedText[pos];
            if(currStart == jumpSize-1 && count >= temp)
                break;
            if(temp != 0){
                if(count <= temp)
                    pos += jumpSize;
                else
                    pos += jumpSize-1;
            }else{
                pos += jumpSize;
            }
            count++;
        }
    }

    return decryption;
```

```
}

int main() {

    char patternChar = '-';
    char resetChar = ' ';
    int lineWidth = 90;
    int initialWidth = 50;

    cout << setfill(patternChar) << setw(lineWidth) << patternChar << endl;
    cout << setfill(resetChar);
    cout << setw(initialWidth) << "Railfence Cipher" << endl;
    cout << setfill(patternChar) << setw(lineWidth) << patternChar << endl;
    cout << setfill(resetChar);


    string text {};
    cout << "Enter text: " << endl;
    getline(cin, text);

    size_t depth;
    cout << "Enter depth (Depth must be greater than 0): ";
    cin >> depth;


```

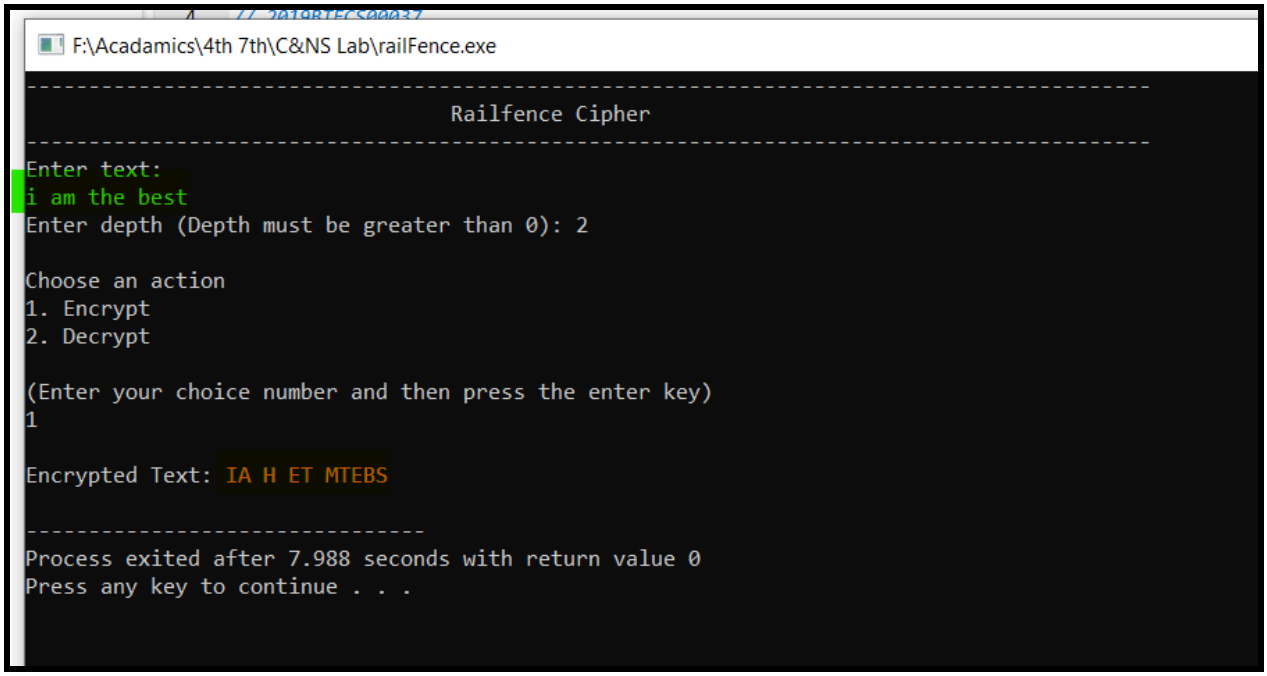
```
    cout << "\nChoose an action" << endl;
    cout << "1. Encrypt" << endl;
    cout << "2. Decrypt" << endl;
    cout << "\n(Enter your choice number and then press the enter key)" << endl;

    int choice;
    cin >> choice;

    if(choice == 1){
        string encryptedText = encrypt(text, depth);
        cout << "\nEncrypted Text: " << encryptedText << endl;
    }else if(choice == 2){
        string decryptedText = decrypt(text, depth);
        cout << "\nDecrypted Text: " << decryptedText << endl;
    }else {
        cout << "Invalid choice..." << endl;
        cout << "Program Terminated." << endl;
    }

    return 0;
}
```

Output


F:\Acadamics\4th 7th\C&NS Lab\railFence.exe
Railfence Cipher
Enter text:
i am the best
Enter depth (Depth must be greater than 0): 2
Choose an action
1. Encrypt
2. Decrypt
(Enter your choice number and then press the enter key)
1
Encrypted Text: IA H ET MTEBS
Process exited after 7.988 seconds with return value 0
Press any key to continue . . .