C&NS Lab Assignment 5

Onkar Santosh Gavali (2019BTECS00037)

Batch B2

Index

Columnar encryption

More complex approach to encrypting plain text. We need a key for rearranging columns. The plaintext is written in rectangle row by row and read column by column for encryption

Eg “the world is big”

1	2	3	4
t	h	e	w
o	r	l	d
i	s	b	i
g			

Cipher text: TOIGHRSELBWDI

Code

railFence.cpp columnarCipher.cpp

columnarCipher.cpp

```cpp
// Cryptography and Network Security Lab
// Assignment 5
// Onkar Gavali
// 2019BTECS00037
// Batch B2
// This program implements the Columnar Cipher
#include <iostream>
#include <map>
#include <vector>
#include <iomanip>
using namespace std;

char upper(char c){
    if(c>='a' && c<='z') return 'A'+c-'a';
    return c;
}
string encrypt(string plainText, string key){
    string cipherText{};
    map<char,string> columnar;

    for(size_t i = 0, j=0; i < plainText.size(); i++, j=(j+1)%key.size()){
        if(plainText[i]==' '){
            j--;
            continue;
        }
        columnar[key[j]]+=upper(plainText[i]);
    }
    for(auto a:columnar){
        cipherText+=a.second;
    }
    return cipherText;
}

int main() {

    string text{};
    string key{};

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

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

    cout << "Enter text: ";
```

```
51      getline(cin, text);
52
53      cout << " Encryption:" << endl;
54
55      if(choice == 1){
56          string encryptedText = encrypt(text, key);
57          cout << "\nEncrypted Text: " << encryptedText << endl;
58      }else {
59          cout << "Invalid choice..." << endl;
60          cout << "Program Terminated." << endl;
61      }
62
63      return 0;
64  }
```

Output