Name: Tanmay Vig

Roll No: 19BCS061

Class: 3rd year(5th sem)

Lab 7: WAP to encrypt and decrypt Plain text using Rail-fence cipher.

Answer:

Source code:

```
#include<iostream>

#include<cstring>

#include<string>

using namespace std;


// encryption function

string encrypt(string text, int key)

{

    // 2d array for encrypting of size key,test size

    char rail[key][(text.length())];


    // filling elements with \t

    for (int i=0; i < key; i++)

        for (int j = 0; j < text.length(); j++)

            rail[i][j] = '\t';


    bool down = false; //variable for direction

    int row = 0, col = 0;
```

```
    for (int i=0; i < text.length(); i++)
    {
        if (row == 0 || row == key-1)
            down = !down;// reversing direction when reached end

        rail[down?row++ : row--][col++] = text[i];// stuffing element with text to be encrypted


    }


    string result;

    // traversing in matrix to create cypher
    for (int i=0; i < key; i++)
        for (int j=0; j < text.length(); j++)
            if (rail[i][j]!='\t')
                result.push_back(rail[i][j]);

    return result;
}

// decryption function
string decrypt(string cipher, int key)
{
    // 2d array for encrypting of size key,test size
```

```
char rail[key][cipher.length()];


// filling elements with \t
for (int i=0; i < key; i++)
    for (int j = 0; j < cipher.length(); j++)
        rail[i][j] = '\t';


bool down; // variable for direction
int row = 0, col = 0;


// pointing elements to be stuffed
for (int i=0; i < cipher.length(); i++)
{
    if (row == 0)
        down = true;
    if (row == key-1)
        down = false;


    rail[down?row++: row--][col++] = '\n';
}


int index = 0;
//stuffing elements with cipher
for (int i=0; i<key; i++)
    for (int j=0; j<cipher.length(); j++)
        if (rail[i][j] == '\n' && index<cipher.length())
            rail[i][j] = cipher[index++];
```

```
    string result;

    row = 0, col = 0;
    // traversing in zig-zag order to Decrypt
    for (int i=0; i< cipher.length(); i++)
    {

        if (row == 0)
            down = true;
        if (row == key-1)
            down = false;



        if (rail[row][col] != '*')
            result.push_back(rail[down?row++: row--][col++]);



    }
    return result;
}

int main()
{
    string in; int k,choose;
    cout<<"Enter Text to encrypt: ";
    getline(cin,in);// taking input text
```

```
    cout<<"Enter key: ";
    cin>>k; // taking input key
    cout <<"Encrypted Text: " <<(in=encrypt(in, k)) << endl;// encrypting text, assigning and printing cipher

    cout<<"Do you want to decrypt? if yes then Enter 1 else 0\n";
    cin>>choose;
    if(choose){
        cout << "Decrypted Text:" <<decrypt(in,k) << endl;// decrypting cipher.
    }

    return 0;
}
```

Output: