Pandit Deendayal Energy University

School of Technology

Information Security Lab

B.Tech-Computer Science & Engineering (Sem-V)

PATEL VEDANT H.

19BCP138

DIVISION – 2

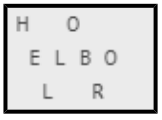
Lab 3 Assignment

- **Aim:** Study and Implement program for Rail Fence Cipher.

❖ Introduction:

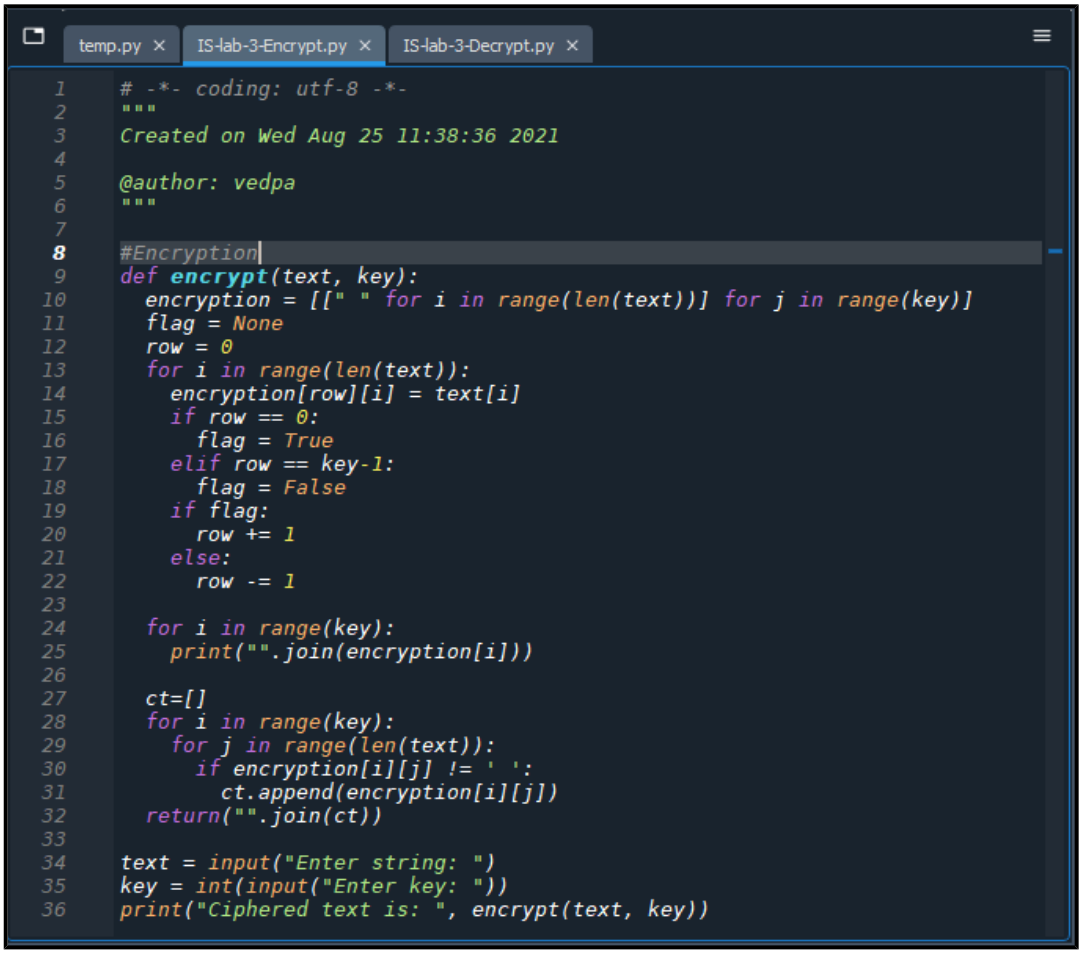
In a fence cipher, letters aren't changed, only switched around regarding their positioning within the message. This kind of cipher is called a transposition cipher because letters are simply transposed in terms of their placement. Transposition ciphers are just like the fencing cipher and it is a relatively weak type of encoding, and may easily be broken, especially with today's technology. These forms of ciphers go back to American warfare, where soldiers would use the code to send encrypted messages.

In a fence cipher, the author takes a message and writes it into descending lines or rails. The fence cipher is usually called a zig-zag cipher if the author uses a zigzag or W pattern to represent text. To encode the text, the user takes the letters within the top line, or rail, and puts them together. And then writes the second line and also the third line. The result is an encoded line of text. For instance, using the phrase "HELLO BRO" and a series of "three" rails, the result (for a linear descent) would be HOELB OLR.



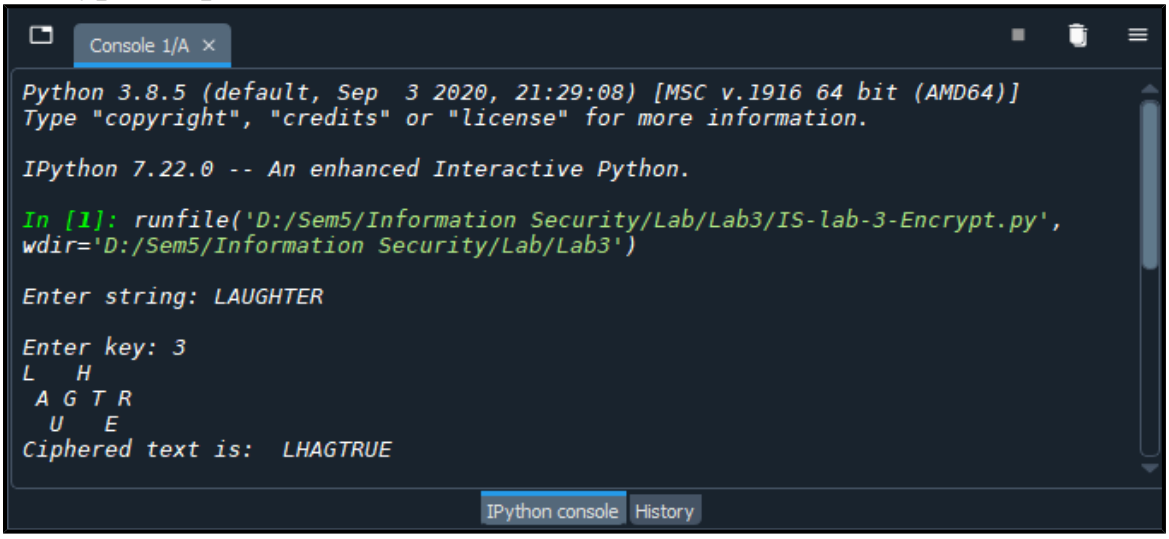
❖ Program:

➢ Encrypt Python Program:



```
# -*- coding: utf-8 -*-
"""
Created on Wed Aug 25 11:38:36 2021

@author: vedpa
"""

#Encryption
def encrypt(text, key):
  encryption = [[" " for i in range(len(text))] for j in range(key)]
  flag = None
  row = 0
  for i in range(len(text)):
    encryption[row][i] = text[i]
    if row == 0:
      flag = True
    elif row == key-1:
      flag = False
    if flag:
      row += 1
    else:
      row -= 1

  for i in range(key):
    print("".join(encryption[i]))

  ct=[]
  for i in range(key):
    for j in range(len(text)):
      if encryption[i][j] != ' ':
        ct.append(encryption[i][j])
  return("".join(ct))

text = input("Enter string: ")
key = int(input("Enter key: "))
print("Ciphered text is: ", encrypt(text, key))
```

➢ Encrypt Output:



```
Python 3.8.5 (default, Sep  3 2020, 21:29:08) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.22.0 -- An enhanced Interactive Python.

In [1]: runfile('D:/Sem5/Information Security/Lab/Lab3/IS-lab-3-Encrypt.py',
wdir='D:/Sem5/Information Security/Lab/Lab3')

Enter string: LAUGHTER

Enter key: 3
L   H
 A G T R
  U   E
Ciphered text is:  LHAGTRUE
```

➢ Decrypt Python Program:

D:\Sem5\Information Security\Lab\Lab3\IS-lab-3-Decrypt.py

temp.py × | IS-lab-3-Encrypt.py × | IS-lab-3-Decrypt.py ×

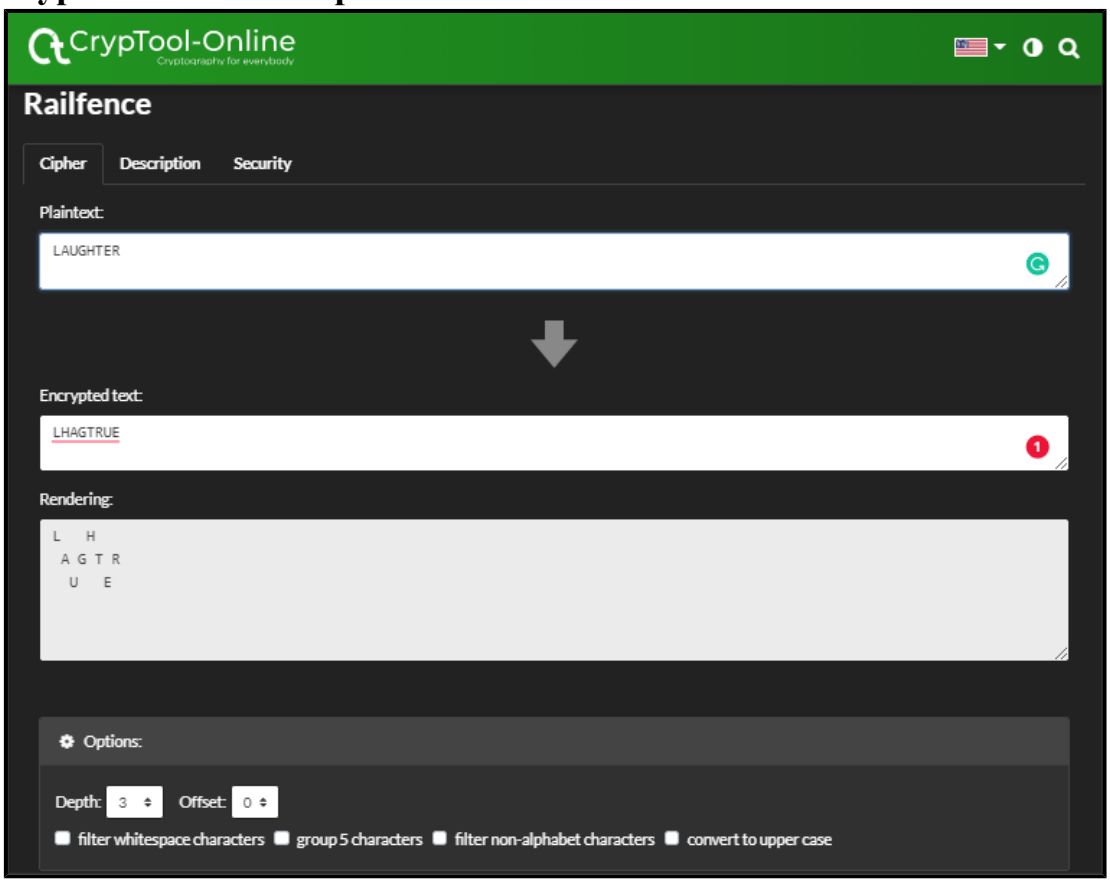
```python
# -*- coding: utf-8 -*-
"""
Created on Wed Aug 25 11:38:35 2021

@author: vedpa
"""

#Decryption
def decrypt(s, k):
  decryption=[[" " for i in range(len(s))] for j in range(k)]
  flag= None
  row, col = 0, 0

  for i in range(len(s)):
    if row==0:
      flag=True
    elif row==k-1:
      flag=False
    decryption[row][col]= "*"
    col +=1

    if flag:
      row+=1
    else:
      row-=1
  for i in range(k):
    print("".join(decryption[i]))

  index = 0
  for i in range(k):
    for j in range(len(s)):
      if ((decryption[i][j]=='*') and (index < len(s))):
        decryption[i][j] = s[index]
        index += 1

  dt=[]
  row, col = 0, 0
  for j in range(len(s)):
    if row==0:
      flag=True
    if row==k-1:
      flag=False

    if decryption[row][col] !='*':
      dt.append(decryption[row][col])
      col += 1

    if flag:
      row+=1
    else:
      row-=1

  return("".join(dt))

s=input("Enter string: ")
k=int(input("Enter key: "))
print("Deciphered text is: ", decrypt(s, k))
```

➢ Decrypt Output:

Console 1/A ×

```
In [2]: runfile('D:/Sem5/Information Security/Lab/Lab3/IS-lab-3-Decrypt.py',
wdir='D:/Sem5/Information Security/Lab/Lab3')

Enter string: LHAGTRUE

Enter key: 3
*   *
 * * * *
  *   *
Deciphered text is:  LAUGHTER
```

➢ **CrypTool Online Output:**



❖ Cryptanalysis:

The cipher's key is the number of rails. It is understood, the cipher-text will be decrypted by using the above algorithm. The length of the cipher-text isn't usable. The length of cipher-text is the same because of the plaintext. Therefore, the amount of usable keys is low, allowing the brute-force attack of trying all possible keys. As a result, the rail-fence cipher is taken into account weak.

❖ Applications:

- It was used by the Greeks, who created a special tool, called *scytale*, to make message encryption and decryption easier.
- Currently, it is usually used with a piece of paper. The letters are arranged in a way that is similar to the shape of the top edge of the rail fence.

❖ Reference:

- https://en.wikipedia.org/wiki/Rail_fence_cipher
- https://www.techopedia.com/definition/29767/rail-fence-cipher

- http://www.crypto-it.net/eng/simple/rail-fence-cipher.html#:~:text=The%20Rail%20Fence%20Cipher%20was,with%20a%20piece%20of%20paper