Expt no. 3

Aim - To design & implement a product cipher using substitution cipher & column or transposition cipher

Theory -

1. Caesar Cipher

The Caesar cipher technique is one of the easiest and simple method of encryption technique. It's a type of substitution cipher where each letter of a given text is replaced by a letter some fixed number of positions down the alphabet.

The method is named after Julius Caesar who apparently used it to communicate with his officials.

Encryption

Encryption can be represented using modular arithmatic by first transforming letters into numbers according to scheme A=0, B=1, ... Z=25.

Encryption of a letter by shift n can be mathematically represented as

$$E_n(x) = (x+n) \bmod 26$$

Encryption phase with shift n

$$D_n(x) = (x-n) \bmod 26$$

Decryption phase with shift n

Page No.
Date

Columnar Transposition cipher
It is a form of transposition cipher just Rail Fence. It involves writing plaintext out in rows and reading ciphertext off column in one by one.

Encryption
① message is written out in rows of fixed length & read out again column by column
② Width of rows & permutation of columns are usually defined by a keyword.
③ Any empty spaces are filled by bogus characters
④ Finally message is read in column by column in order specified by keyword.

Decryption
① Recipient has to work out the column length by dividing message length by key length
② Write message in columns again then re-order the columns reforming keyword.

- Hill Cipher

Apdygraphic Substitution Cipher based on linear algebra Each letter is represented by a number modulo 26. To encrypt a message, each block of n letters is multiplied by an invertible $n \times n$ matrix against modulus 26. To decrypt the message each block is multiplied by inverse of matrix used for encryption.

CAESAR CIPHER

PROGRAM :

```
def encrypt(text,s):
    result = ""
    for i in range(len(text)):
        char = text[i]

        if (char.isupper()):
            result += chr((ord(char) + s-65) % 26 + 65)
        else:
            result += chr((ord(char) + s - 97) % 26 + 97)

    return result
text=input("Enter Plaintext:")
s=int(input("Enter the number of shifts:"))
print("Ciphertext:" + encrypt(text,s))
```

OUTPUT :

```
PS C:\DesktopLite\sem 5\cns\cns lab> python caesarCipher.py
Enter Plaintext:you are not safe now
Enter the number of shifts:5
Ciphertext:dtzsfwjsstysxfkjsstb
PS C:\DesktopLite\sem 5\cns\cns lab>
```

COLUMNAR TRANSPOSITION CIPHER

PROGRAM :

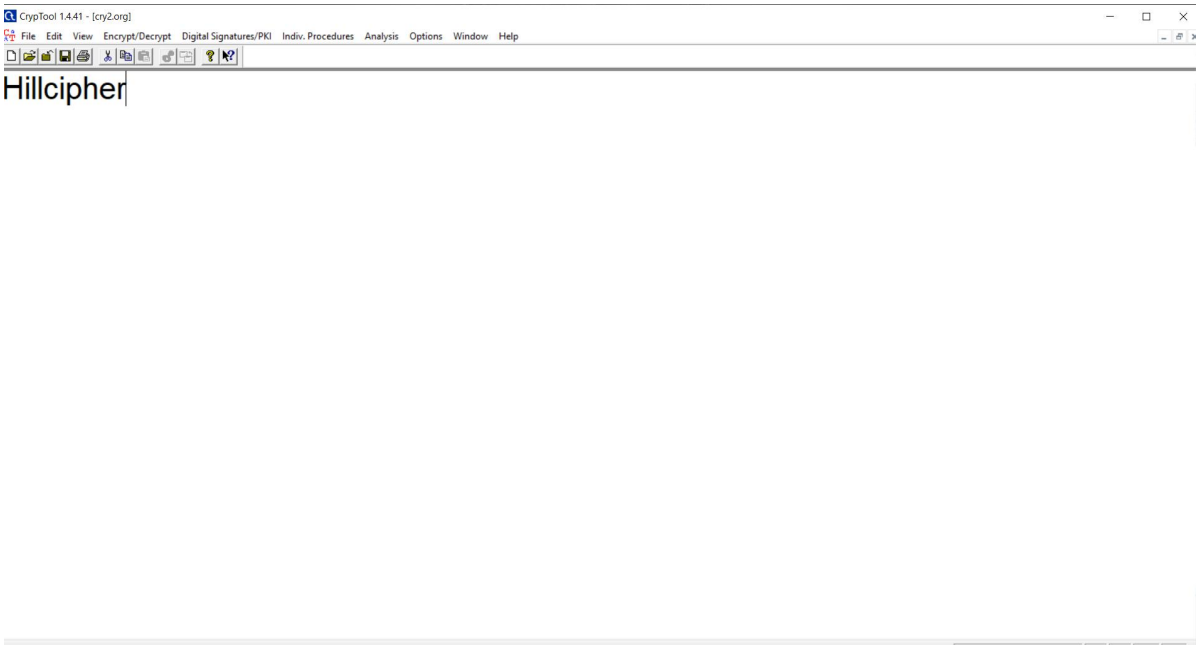
```
import math
key=input("Enter the key:")
def encryptMessage(msg):
    cipher = ""
    k_indx = 0

    msg_len = float(len(msg))
    msg_lst = list(msg)
    key_lst = sorted(list(key))
    col = len(key)

    row = int(math.ceil(msg_len / col))

    fill_null = int((row * col) - msg_len)
    msg_lst.extend('_' * fill_null)

    matrix = [msg_lst[i: i + col]
              for i in range(0, len(msg_lst), col)]

    for _ in range(col):
        curr_idx = key.index(key_lst[k_indx])
```

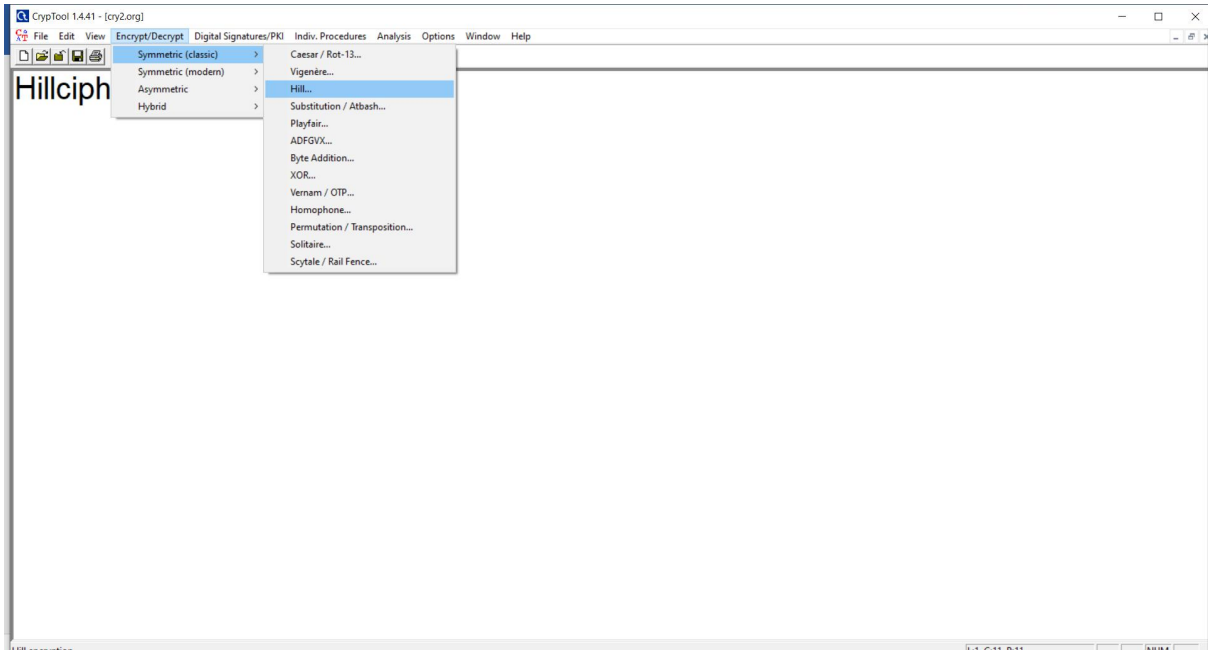
```
        cipher += ''.join([row[curr_idx]
                    for row in matrix])
        k_indx += 1

    return cipher

msg =input("Enter the plaintext:")

cipher = encryptMessage(msg)
print("Encrypted Message: {}".
            format(cipher))
```

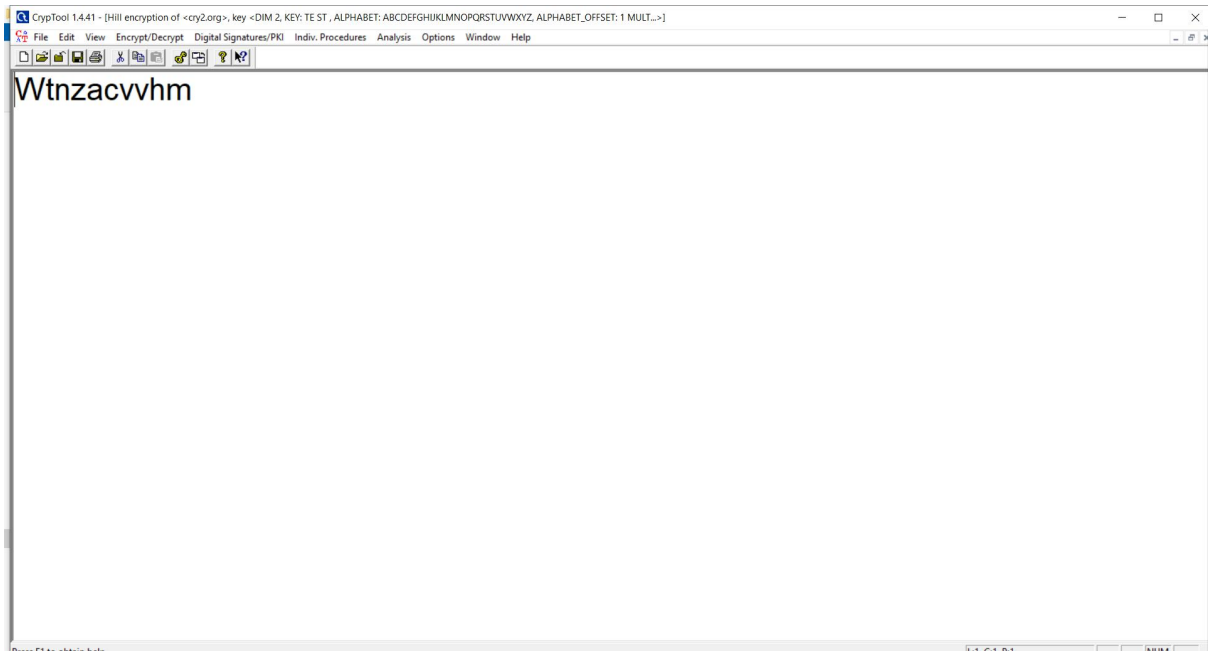
OUTPUT :

```
PS C:\DesktopLite\sem 5\cns\cns lab> python columnarTcipher.py
Enter the key:DANGER
Enter the plaintext:you are not safe
Encrypted Message: o ayesat_ oeunfr
PS C:\DesktopLite\sem 5\cns\cns lab>
```

HILL CIPHER USING CRYPTOOL







Conclusion :

Hence we can conclude that we implemented Caesar cipher and columnar transposition cipher using python and hill cipher in cryptool.