Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

# Experiment 2

Date of Performance : 20-02-2023 Date of Submission: 20-02-2023

SAP Id: 60004200107 Name : Kartik Jolapara

Div: B Batch : B1

## Aim of Experiment

Design and Implement Playfair Cipher. Create two function Encyrpt() and Decyrpt().

(CO1)

## Theory / Algorithm / Conceptual Description

The Playfair cipher uses a 5 by 5 table containing a key word or phrase. Memorization of the keyword and 4 simple rules was all that was required to create the 5 by 5 table and use the cipher.

To generate the key table, one would first fill in the spaces in the table (a modified Polybius square) with the letters of the keyword (dropping any duplicate letters), then fill the remaining spaces with the rest of the letters of the alphabet in order (usually omitting "J" or "Q" to reduce the alphabet to fit; other versions put both "I" and "J" in the same space). The key can be written in the top rows of the table, from left to right, or in some other pattern, such as a spiral beginning in the upper-left-hand corner and ending in the center. The keyword together with the conventions for filling in the 5 by 5 table constitute the cipher key.

To encrypt a message, one would break the message into digrams (groups of 2 letters) such that, for example, "HelloWorld" becomes "HE LL OW OR LD". These digrams will be substituted using the key table. Since encryption requires pairs of letters, messages with an odd number of characters usually append an uncommon letter, such as "X", to complete the final digram. The two letters of the digram are considered opposite corners of a rectangle in the key table. To perform the substitution, apply the following 4 rules, in order, to each pair of letters in the plaintext:

1. If both letters are the same (or only one letter is left), add an "X" after the first letter. Encrypt the new pair and continue. Some variants of Playfair use "Q" instead of "X", but any letter, itself uncommon as a repeated pair, will do.
2. If the letters appear on the same row of your table, replace them with the letters to their immediate right respectively (wrapping around to the left side of the row if a letter in the original pair was on the right side of the row).
3. If the letters appear on the same column of your table, replace them with the letters immediately below respectively (wrapping around to the top side of the column if a letter in the original pair was on the bottom side of the column).

4. If the letters are not on the same row or column, replace them with the letters on the same row respectively but at the other pair of corners of the rectangle defined by the original pair. The order is important – the first letter of the encrypted pair is the one that lies on the same row as the first letter of the plaintext pair.

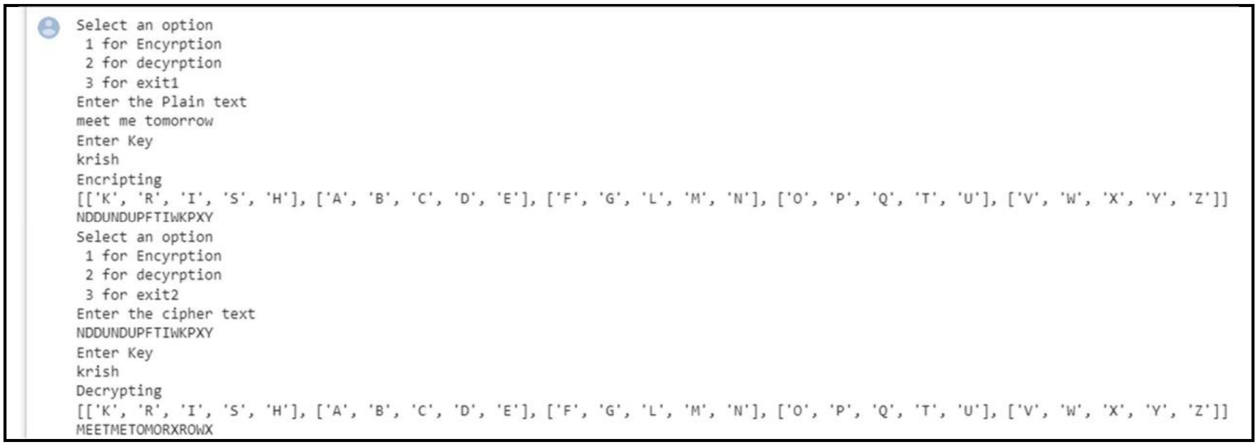
Program

```
def create_matrix(key):     key =
key.upper()
    matrix = [[0 for i in range (5)] for j in range(5)]     letters_added = []     row = 0
col = 0
     for letter in key:          if letter not in letters_added:
matrix[row][col] = letter
letters_added.append(letter)          else:
            continue          if (col==4):
col = 0             row += 1          else:
            col += 1
     for letter in range(65,91):          if letter==74:                    continue
if chr(letter) not in letters_added:
letters_added.append(chr(letter))
                 index = 0      for i in
range(5):          for j in range(5):
            matrix[i][j] = letters_added[index]             index+=1     return
matrix
```

```
def separate_same_letters(message):
    index = 0     while (index<len(message)):
        l1 = message[index]        if index == len(message)-1:            message = message + 'X'
index += 2            continue        l2 = message[index+1]        if l1==l2:            message =
message[:index+1] + "X" + message[index+1:]        index +=2       return message
 def indexOf(letter,matrix):
    for i in range (5):
        try:
            index = matrix[i].index(letter)            return (i,index)
except:
            continue
 def playfair(key, message, encrypt=True):
    inc = 1     if encrypt==False:
        inc = -1
    matrix = create_matrix(key)     message = message.upper()     message =
message.replace(' ',")        message = separate_same_letters(message)
cipher_text="     for (l1, l2) in zip(message[0::2], message[1::2]):
        row1,col1 = indexOf(l1,matrix)        row2,col2 =
indexOf(l2,matrix)        if row1==row2:
            cipher_text += matrix[row1][(col1+inc)%5] + matrix[row2][(co l2+inc)%5]        elif col1==col2:
            cipher_text += matrix[(row1+inc)%5][col1] + matrix[(row2+inc
)%5][col2]        else:
            cipher_text += matrix[row1][col2] + matrix[row2][col1]     print(matrix)     return
cipher_text
```

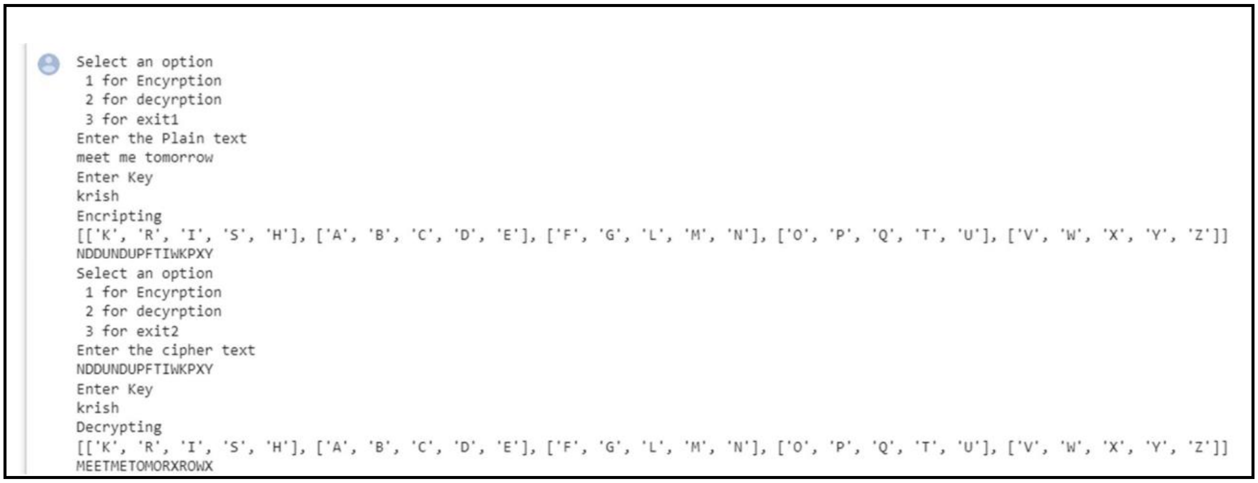
```
option = 1 while(option == 1 or option == 2):
  option = int(input("Select an option\n 1 for Encyrption\n 2 for decyrp tion\n 3 for exit"))   if( option == 1):
    plainText = input("Enter the Plain text\n")     key = input("Enter
Key\n")     print ('Encripting')     print ( playfair(key, plainText))   if(
option == 2):
    cipherText = input("Enter the cipher text\n")     key = input("Enter
Key\n")     print ('Decrypting')     print ( playfair(key, cipherText, False))
if (option == 3):     break
```

## Input



```
Select an option
 1 for Encyrption
 2 for decyrption
 3 for exit1
Enter the Plain text
meet me tomorrow
Enter Key
krish
Encripting
[['K', 'R', 'I', 'S', 'H'], ['A', 'B', 'C', 'D', 'E'], ['F', 'G', 'L', 'M', 'N'], ['O', 'P', 'Q', 'T', 'U'], ['V', 'W', 'X', 'Y', 'Z']]
NDDUNDUPFTIWKPXY
Select an option
 1 for Encyrption
 2 for decyrption
 3 for exit2
Enter the cipher text
NDDUNDUPFTIWKPXY
Enter Key
krish
Decrypting
[['K', 'R', 'I', 'S', 'H'], ['A', 'B', 'C', 'D', 'E'], ['F', 'G', 'L', 'M', 'N'], ['O', 'P', 'Q', 'T', 'U'], ['V', 'W', 'X', 'Y', 'Z']]
MEETMETOMORXROWX
```

## Output



```
Select an option
 1 for Encyrption
 2 for decyrption
 3 for exit1
Enter the Plain text
meet me tomorrow
Enter Key
krish
Encripting
[['K', 'R', 'I', 'S', 'H'], ['A', 'B', 'C', 'D', 'E'], ['F', 'G', 'L', 'M', 'N'], ['O', 'P', 'Q', 'T', 'U'], ['V', 'W', 'X', 'Y', 'Z']]
NDDUNDUPFTIWKPXY
Select an option
 1 for Encyrption
 2 for decyrption
 3 for exit2
Enter the cipher text
NDDUNDUPFTIWKPXY
Enter Key
krish
Decrypting
[['K', 'R', 'I', 'S', 'H'], ['A', 'B', 'C', 'D', 'E'], ['F', 'G', 'L', 'M', 'N'], ['O', 'P', 'Q', 'T', 'U'], ['V', 'W', 'X', 'Y', 'Z']]
MEETMETOMORXROWX
```