

Name - Khushi Chhatwani

Roll no. csc/21/55

Course -B.Sc(H) CS

Ques 9. Implement product cipher transposition operation.

Ans:-

```
def product_cipher_transposition(plaintext, key):
    key_length = len(key)
    plaintext_length = len(plaintext)
    if plaintext_length % key_length != 0:
        padding_length = key_length - (plaintext_length % key_length)
        plaintext += ' ' * padding_length
        plaintext_length += padding_length
    blocks = [plaintext[i:i+key_length]
               for i in range(0, plaintext_length, key_length)]
    transposed_blocks = []
    for block in blocks:
        transposed_block = [None] * key_length
        for i, j in enumerate(key):
            transposed_block[j] = block[i]
        transposed_blocks.append(''.join(transposed_block))
    return ciphertext

if __name__ == "__main__":
    plaintext = input("Enter the message : ")
    key = (2, 0, 1)
    ciphertext = product_cipher_transposition(plaintext, key)
    print(ciphertext)
```

Output

```
def product_cipher_transposition(plaintext, key):
    key_length = len(key)
    plaintext_length = len(plaintext)
    if plaintext_length % key_length != 0:
        padding_length = key_length - (plaintext_length % key_length)
        plaintext += ' ' * padding_length
        plaintext_length += padding_length

    blocks = [plaintext[i:i+key_length] for i in range(0, plaintext_length, key_length)]

    transposed_blocks = []
    for block in blocks:
        transposed_block = [None] * key_length
        for i, j in enumerate(key):
            transposed_block[j] = block[i]
        transposed_blocks.append(''.join(transposed_block))

    ciphertext = ''.join(transposed_blocks)
    return ciphertext

if __name__ == "__main__":
    plaintext = input("Enter the message : ")
    key = (2, 0, 1)
    ciphertext = product_cipher_transposition(plaintext, key)
    print(ciphertext)

Enter the message : come at taj 8:00pm
omc ae ttj a:08pm0
```