**AIM: Implement Product Cipher**

CODE:

```
def enc_substitution(text, k):
        cipher = ''
        for char in text:
                if char == ' ':
                  cipher += char
                elif char.isupper():
                  cipher += chr((ord(char) + k - 65) % 26 + 65)
                else:
                  cipher += chr((ord(char) + k - 97) % 26 + 97)
        return cipher


def dec_substitution(text, k):
        p = ''
        for char in text:
                if char == ' ':
                  p += char
                elif char.isupper():
                  p += chr((ord(char) - k - 65) % 26 + 65)
                else:
                  p += chr((ord(char) - k - 97) % 26 + 97)
        return p


def enc_rail_fence(text, k):
        rail = [""] * k
        layer = 0
        for t in cipher:
           rail[layer] += t
           #print(rail)
           if layer >= k - 1:
              layer = 0
```

```
            else:
                layer += 1
        return ''.join(rail)


def create_rail(text, k):
        rail = [""] * k
        layer = 0
        for t in text:
            rail[layer] += t
            #print(rail)
            if layer >= k - 1:
                layer = 0
            else:
                layer += 1

        return rail


def enc_rail_fence(text, k):
        rail = create_rail(text, k)
        return ''.join(rail)


def dec_rail_fence(text, k):
        t = 'A' * len(text)
        rail = create_rail(t, k)
        for i in range(len(rail)):
                rail[i] = rail[i].replace(rail[i], text[0:len(rail[i])])
                text = text[len(rail[i]):]

        count = 0
        dec = ''
        for i in range(k):
                for j in range(k):
                        if count == len(t):
```

```
                    break
                dec += rail[j][i]

                count += 1
            else:
                continue
            break

    return dec


def product_cipher(text, k):
    cipher = enc_substitution(text, k)
    encrypted = enc_rail_fence(cipher, k)

    print("Encrypted: ", encrypted)

    dec_rail = dec_rail_fence(encrypted, k)
    decrypted = dec_substitution(dec_rail, k)

    print("Decrypted: ", decrypted)


if __name__ == '__main__':
    text = 'Bhushan'
    print('Plain text: ', text)
    k = 5
    product_cipher(text, k)
```

OUTPUT:

```
C:\Users\Bhushan Borole\Desktop\Coding\Sem6\CSS\product_cipher>python product_cipher.py
Plain text:  Bhushan
Encrypted:  Gfmszxm
Decrypted:  Bhushan
```