Practical-3

Aim : Alice wants to send some confidential information to Bob over a secure network, you have to create perform following task :
1)Provide Security using Caesar Cipher Algorithm
2)Find the all possible Cipher Text & Plaintext pairs
3)Provide Security Mono-alphabetic Cipher Algorithm

Caesar Cipher

```
def encrypt(plain_text, key):
    cipher_text = ""
    for char in plain_text:
        if char.isalpha():
            ascii_offset = ord('a')
            encrypted_char = chr((ord(char.lower()) - ascii_offset + key) % 26
+ ascii_offset)
            cipher_text += encrypted_char
        else:
            cipher_text += char
    return cipher_text

plain_text = input("Enter the plain text: ").lower()
key = int(input("Enter the key: "))

cipher_text = encrypt(plain_text, key)
print("Cipher text:", cipher_text)
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR

PS E:\B.Tech_SEM-6_CBA\INS> & C:/Users/panjw/AppData/Local/Programs/Python/Python311/python.exe "e:/B.Tech_SEM
-6_CBA/INS/Practical-3/Caesar Cipher.py"
Enter the plain text: aniket
Enter the key: 5
Cipher text: fsnpjy
PS E:\B.Tech_SEM-6_CBA\INS>
```

Find the all possible Cipher Text & Plaintext pairs

```
def decrypt(cipher_text, key):
    plaintext = ""
    for char in cipher_text:
        if char.isalpha():
            ascii_val = ord(char)
            decrypted_val = (ascii_val - key) % 26
            decrypted_char = chr(decrypted_val + ord('A'))
            plaintext += decrypted_char
        else:
```

```
            plaintext += char
    return plaintext

def find_all_plaintexts(cipher_text):
    all_plaintexts = []
    for key in range(26):
        plaintext = decrypt(cipher_text, key)
        all_plaintexts.append(plaintext)
    return all_plaintexts

cipher_text = input("Enter the cipher text: ")
plaintexts = find_all_plaintexts(cipher_text)
print("All possible plaintexts:")
for plaintext in plaintexts:
    print(plaintext)
```

output :-

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERRO

-6_CBA/INS/Practical-3/Caesar Cipher2.py"
Enter the cipher text: fsnpjy
All possible plaintexts:
YLGICR
XKFHBQ
WJEGAP
VIDFZO
UHCEYN
TGBDXM
SFACWL
REZBVK
QDYAUJ
PCXZTI
OBWYSH
NAVXRG
MZUWQF
LYTVPE
KXSUOD
JWRTNC
IVQSMB
HUPRLA
GTOQKZ
FSNPJY
ERMOIX
DQLNHW
CPKMGV
BOJLFU
ANIKET
ZMHJDS
PS E:\B.Tech_SEM-6_CBA\INS>
```

Provide Security Mono-alphabetic Cipher Algorithm

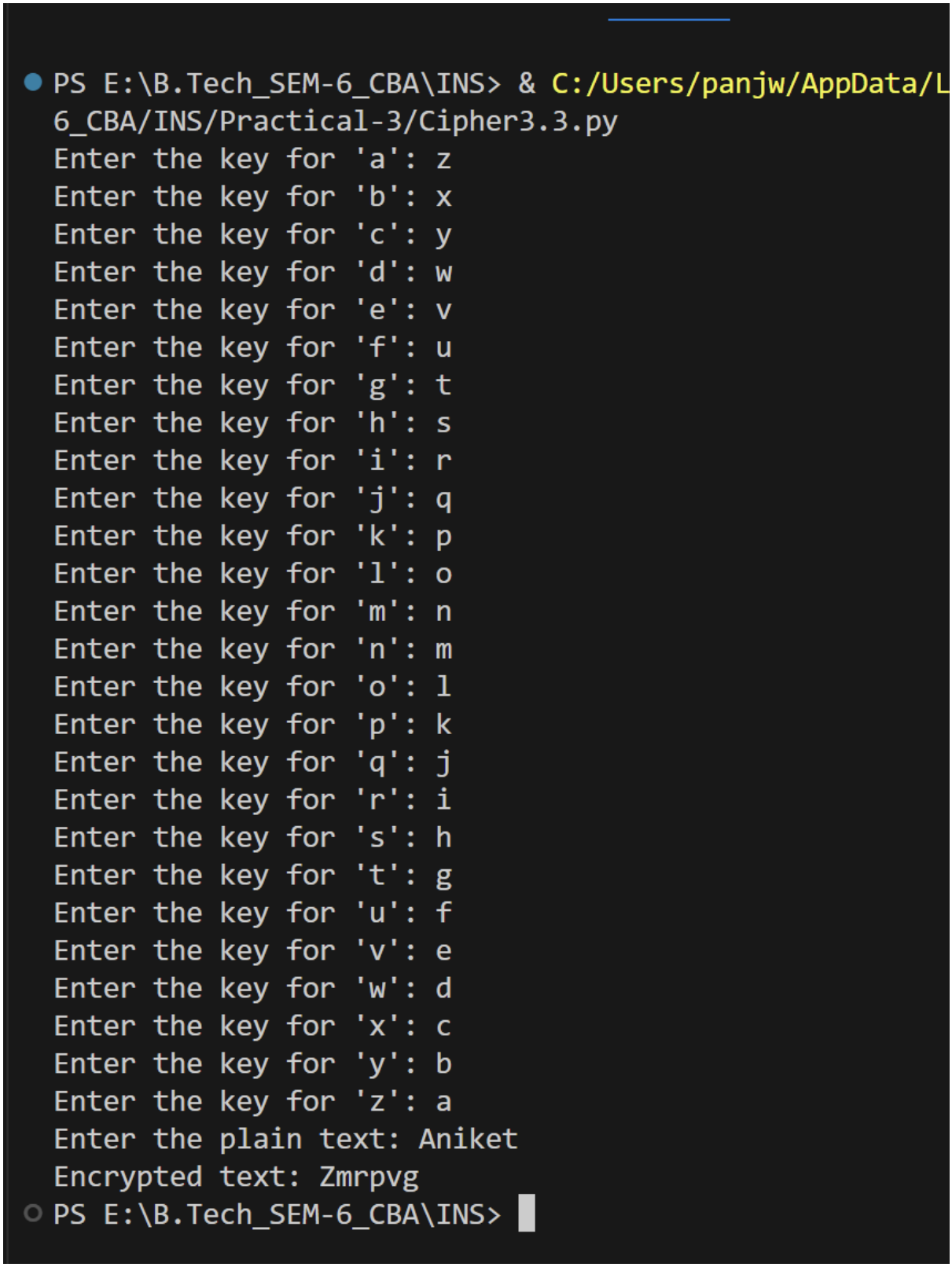
```
def encrypt(plain_text, key):
    cipher_text = ""
    for char in plain_text:
        if char.isalpha():
            if char.isupper():
                index = ord(char) - ord('A')
                cipher_char = key[index].upper()
            else:
                index = ord(char) - ord('a')
                cipher_char = key[index].lower()
            cipher_text += cipher_char
        else:
            cipher_text += char
    return cipher_text

def decrypt(cipher_text, key):
    plain_text = ""
    for char in cipher_text:
        if char.isalpha():
            if char.isupper():
                index = ord(char) - ord('A')
                plain_char = chr(key[index].upper())
            else:
                index = ord(char) - ord('a')
                plain_char = chr(key[index].lower())
            plain_text += plain_char
        else:
            plain_text += char
    return plain_text

def main():
    key = []
    for i in range(26):
        key.append(input(f"Enter the key for '{chr(ord('a')+i)}': "))
    plain_text = input("Enter the plain text: ")
    cipher_text = encrypt(plain_text, key)
    print("Encrypted text:", cipher_text)


if __name__ == "__main__":
    main()
```

output: -


PS E:\B.Tech_SEM-6_CBA\INS> & C:/Users/panjw/AppData/L
6_CBA/INS/Practical-3/Cipher3.3.py
Enter the key for 'a': z
Enter the key for 'b': x
Enter the key for 'c': y
Enter the key for 'd': w
Enter the key for 'e': v
Enter the key for 'f': u
Enter the key for 'g': t
Enter the key for 'h': s
Enter the key for 'i': r
Enter the key for 'j': q
Enter the key for 'k': p
Enter the key for 'l': o
Enter the key for 'm': n
Enter the key for 'n': m
Enter the key for 'o': l
Enter the key for 'p': k
Enter the key for 'q': j
Enter the key for 'r': i
Enter the key for 's': h
Enter the key for 't': g
Enter the key for 'u': f
Enter the key for 'v': e
Enter the key for 'w': d
Enter the key for 'x': c
Enter the key for 'y': b
Enter the key for 'z': a
Enter the plain text: Aniket
Encrypted text: Zmrpvg
PS E:\B.Tech_SEM-6_CBA\INS>