**Affine Cipher**

def affine\_encrypt(text, key):

    # C = (a\*P + b) % 26

    encrypted\_text = ''

    for t in text:

        if t!= " ":

            if t.islower():

                char = ((key[0]\*(ord(t) - ord('a')) + key[1]) % 26) + ord('a')

                encrypted\_text+= chr(char)

            elif t.isupper():

                char = ((key[0]\*(ord(t) - ord('A')) + key[1]) % 26) + ord('A')

                encrypted\_text+= chr(char)

        elif t==" ":

            encrypted\_text+=" "

    return encrypted\_text

# affine cipher decryption function

def affine\_decrypt(cipher, key):

    # P = (a^-1 \* (C - b)) % 26

    decrypted\_text = ''

    key\_inv = 0

    for i in range(26):

        flag = (key[0]\* i) % 26

        if (flag == 1):

            key\_inv = i

    for t in cipher:

        if t!= " ":

            if t.islower():

                char = (key\_inv\*(ord(t) - ord('a') - key[1]) % 26) + ord('a')

                decrypted\_text+= chr(char)

            elif t.isupper():

                char = (key\_inv\*(ord(t) - ord('A') - key[1]) % 26) + ord('A')

                decrypted\_text+= chr(char)

        elif t==" ":

            decrypted\_text+=" "

    return decrypted\_text

def main():

    print("Affine Cipher")

    text = input("Enter the text: ")

    k1 = int(input("Enter first key: "))

    k2 = int(input("Enter second key: "))

    key = [k1, k2]

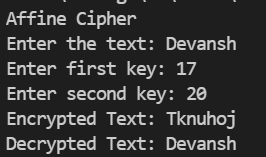
    affine\_encrypted\_text = affine\_encrypt(text, key)

    print('Encrypted Text: '+ affine\_encrypted\_text)

    print('Decrypted Text: '+ affine\_decrypt(affine\_encrypted\_text, key))

main()

**Output:**



**Vignere Cipher**

def keyGenerationFunc(text, key):

    key = list(key)

    if len(text) == len(key):

        return key

    else:

        for i in range(len(text) - len(key)):

            key.append(key[i % len(key)])

    return "".join(key)

def vignere\_encrypt(text, key):

    cipherText = []

    for i in range(len(text)):

        temp = ((ord(text[i]) + ord(key[i])) % 26) + 65

        cipherText.append(chr(temp))

    return "".join(cipherText)

def vignere\_decrypt(cipherText, key):

    originalText = []

    for i in range(len(cipherText)):

        x = ((ord(cipherText[i]) - ord(key[i]) + 26) % 26) + 65

        originalText.append(chr(x))

    return("" . join(originalText))

print("Vignere Cipher")

text = input("Enter text: ").upper()

keyword = input("Enter Key: ")

key = keyGenerationFunc(text, keyword)

cipherText = vignere\_encrypt(text,key)

print("Cipher text :", cipherText)

print("Decrypted Text :",vignere\_decrypt(cipherText, key))

**Output:**

