program for one-time pad version of the Vigenère cipher. In this scheme, the key is a stream of random numbers between 0 and 26. For example, if the key is 3 19 5 . . . , then the first letter of plaintext is encrypted with a shift of 3 letters, the second with a shift of 19 letters, the third with a shift of 5 letters, and so on.

import random

def generate\_key(length):

"""Generate a random key of numbers between 0 and 25."""

return [random.randint(0, 25) for \_ in range(length)]

def encrypt(plaintext, key):

ciphertext = ""

for i, char in enumerate(plaintext.upper()):

if char.isalpha():

shift = key[i]

encrypted\_char = chr((ord(char) - 65 + shift) % 26 + 65)

ciphertext += encrypted\_char

else:

ciphertext += char # keep spaces/punctuation as is

return ciphertext

def decrypt(ciphertext, key):

plaintext = ""

for i, char in enumerate(ciphertext.upper()):

if char.isalpha():

shift = key[i]

decrypted\_char = chr((ord(char) - 65 - shift) % 26 + 65)

plaintext += decrypted\_char

else:

plaintext += char

return plaintext

# Example usage

plaintext = input("Enter plaintext: ").upper()

key = generate\_key(len(plaintext))

print("Random Key (0–25):", key)

ciphertext = encrypt(plaintext, key)

print("Encrypted text:", ciphertext)

decrypted\_text = decrypt(ciphertext, key)

print("Decrypted text:", decrypted\_text) 