**Ques 10.** . Illustrate the Ciphertext only and Known Plaintext attacks

Ans:-

# Bharat\_Sharma CSC/20/50 Univ\_Roll\_No: 20059570040

import random

import string

alphabet = list(string.ascii\_lowercase)

cipher\_key = alphabet.copy()

random.shuffle(cipher\_key)

cipher\_key = ''.join(cipher\_key)

plaintext = "the quick brown fox jumps over the lazy dog"

ciphertext = ""

for letter in plaintext:

if letter.isalpha():

ciphertext += cipher\_key[alphabet.index(letter.lower())]

else:

ciphertext += letter

print("Original plaintext:", plaintext)

print("Encrypted ciphertext:", ciphertext)

print("Cipher key:", cipher\_key)

frequencies = {}

for letter in ciphertext:

if letter.isalpha():

if letter.lower() in frequencies:

frequencies[letter.lower()] += 1

else:

frequencies[letter.lower()] = 1

sorted\_frequencies = sorted(frequencies.items(), key=lambda x: x[1], reverse=True)

most\_frequent = [x[0] for x in sorted\_frequencies]

print("Most frequent letters in ciphertext:", most\_frequent)

known\_plaintext = "the"

matching\_pairs = []

for i in range(len(known\_plaintext)):

plaintext\_letter = known\_plaintext[i]

ciphertext\_letter = ciphertext[i]

matching\_pairs.append((plaintext\_letter, ciphertext\_letter))

matching\_pairs = sorted(matching\_pairs, key=lambda x: x[1])

matching\_key = ""

for pair in matching\_pairs:

matching\_key += alphabet[cipher\_key.index(pair[1].lower())]

print("Matching pairs:", matching\_pairs)

print("Matching key:", matching\_key)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_OUTPUT:-

