**Name:** Arham Sharif

**Seat No.:** EB21102022

**Section:** B

**Subject:** Network Security & Cryptography

**Language:** Python

**LAB# 1**

**CESEAR CIPHER**

**Objective:** Design and implement a simple encoding and decoding program in Python that allows users to input a string, encode it using a specified algorithm, and then decode it back to the original form. This lab aims to reinforce understanding of string manipulation, algorithmic concepts, and basic programming skills while demonstrating the principles of encoding and decoding.

**Technique:** Substitution

**Code:**

"""-----------------ENCODE------------------"""

# Function Can Encode Char

def encodeChar(char):

encode\_char = ''

index = -1

for i in range(len\_char\_arr):

if char\_arr[i] == char:

index = i + inc

if index >= len\_char\_arr:

index %= len\_char\_arr

encode\_char = char\_arr[index]

break

if index != -1:

return encode\_char

else:

return char

if \_\_name\_\_ == "\_\_main\_\_":

# Initialize Variables

inc = 3

char\_arr = [chr(i) for i in range(256)]

len\_char\_arr = len(char\_arr)

encoded\_arr = []

# Input

decoded\_text = input("Enter Text To Encode: ")

for char in decoded\_text:

# Call Function -> Encode Char

encoded\_arr.append(encodeChar(char))

# Print Encoded Text

print(f'Encode Text: {"".join(encoded\_arr)}')

**Output:**



**Code:**

"""-----------------DECODE------------------"""

# Function to Decode Char

def decodeChar(char):

decode\_char = ''

index = -1

for i in range(len\_char\_arr):

if char\_arr[i] == char:

index = i - inc

if index < 0:

index += len\_char\_arr

decode\_char = char\_arr[index]

break

if index != -1:

return decode\_char

else:

return char

if \_\_name\_\_ == "\_\_main\_\_":

# Initialize Variables

inc = 3

char\_arr = [chr(i) for i in range(256)]

len\_char\_arr = len(char\_arr)

decoded\_arr = []

# Input

encoded\_text = input("Enter Text To Decode: ")

for char in encoded\_text:

# Call Function -> Decode Char

decoded\_arr.append(decodeChar(char))

# Print Decoded Text

print(f'Decoded Text: {"".join(decoded\_arr)}')

**Output:**

****