

**Experiment 1**

**Date of Performance : 20-02-2023** **Date of Submission: 26-02-2023**

**SAP Id:** 60004200107 **Name :** Kartik Jolapara

**Div:** B **Batch :** B1

**Aim of Experiment**

Design and Implement Encryption and Decryption Algorithm for Caesar cipher cryptographic algorithm by considering letter [A..Z] and digits [0..9]. Create two functions Encrypt() and Decrypt(). Apply Brute Force Attack to reveal secret. Create Function BruteForce().

(CO1)

**Theory / Algorithm / Conceptual Description**

The Caesar cipher works by first choosing a shift value, which is an integer between 1 and 25. This shift value is then used to encode or decode a message. To encode a message, each letter in the message is replaced by the letter that is a certain number of positions down the alphabet. For example, if the shift value is 3, the letter 'A' would be replaced by the letter 'D', 'B' would be replaced by 'E', and so on. To decode a message, the process is simply reversed, by shifting each letter back by the same number of positions.

The algorithm for the Caesar cipher can be summarized as follows:

* Choose a shift value between 1 and 25.
* For each letter in the message:
  + If the letter is uppercase, shift it down the alphabet by the shift value and replace it with the corresponding letter.
  + If the letter is lowercase, shift it down the alphabet by the shift value and replace it with the corresponding letter.
  + If the letter is not a letter (such as a number or symbol), leave it unchanged.
* The resulting message is the encoded message.

To decode a message, the same process is followed, but in reverse, by shifting each letter back up the alphabet by the same number of positions.

**Program**

|  |
| --- |
| def encrypt(message, shift):  ciphertext = ''  for char in message:  # Check if the character is an uppercase or lowercase letter  if char.isupper():  ciphertext += chr((ord(char) + shift - 65) % 26 + 65)  elif char.islower():  ciphertext += chr((ord(char) + shift - 97) % 26 + 97)  else:  ciphertext += char  return ciphertext  def decrypt(ciphertext, shift):  message = ''  for char in ciphertext:  # Check if the character is an uppercase or lowercase letter  if char.isupper():  message += chr((ord(char) - shift - 65) % 26 + 65)  elif char.islower():  message += chr((ord(char) - shift - 97) % 26 + 97)  else:  message += char  return message  def brute\_force\_attack(ciphertext):  for shift in range(1, 26):  message = decrypt(ciphertext, shift)  print(f'Shift = {shift:2d}: {message}')  # Example usage  message = 'This is a secret message'  shift = 5  print("PLAIN TEXT:", message)  print()  ciphertext = encrypt(message, shift)  print("CIPHER TEXT:", ciphertext)  decrypted\_message = decrypt(ciphertext, shift)  print("DECRYPTED TEXT:", decrypted\_message)  print()  brute\_force\_attack(ciphertext) |

**Input**

|  |
| --- |
|  |

**Output**

|  |
| --- |
|  |