Lab Exercise: Basic Data Encryption and Decryption Using Python

Objectives

1. Understand the concept of encryption and decryption.
2. Implement the Caesar Cipher encryption and decryption in Python.
3. Encrypt and decrypt a message using the implemented script.

Requirements

1. Python environment (like IDLE, Jupyter Notebook, or any IDE that supports Python).
2. Basic knowledge of Python programming.

Task Overview

1. Implement a Caesar Cipher encryption and decryption function in Python.
2. Test the functions by encrypting and decrypting a message.

Introduction to Caesar Cipher

The Caesar Cipher is one of the simplest and most widely known encryption techniques. It is a type of substitution cipher where each letter in the plaintext is replaced by a letter some fixed number of positions down the alphabet. This method was named after Julius Caesar, who reportedly used it to communicate with his generals.

Basic Mechanism

Shift/Key: The core of the Caesar Cipher is the shift (also known as the key). It determines how many places each letter of the plaintext is shifted down or up the alphabet.

Encryption Process

If the shift is 3 (a common choice by Caesar himself), then 'A' would be replaced by 'D', 'B' would become 'E', and so forth.

The alphabet is wrapped around, so if the letter is near the end, it goes back to the beginning. For example, with a shift of 3, 'Y' becomes 'B', and 'Z' becomes 'C'.

Decryption Process

To decrypt, you simply shift back by the same number of positions as the encryption shift.

Using our previous example, 'D' would be shifted back to 'A', 'E' to 'B', and so on.

Features and Usage

Simplicity: It is straightforward to understand and implement, making it a great introduction to cryptography.

Historical Significance: While it is named after Julius Caesar, there is evidence that earlier forms of this cipher were used.

Not Secure for Modern Standards: Due to its simplicity, the Caesar Cipher can be easily broken, even without computers. A brute-force attack, trying all possible shifts, is effective since there are only 25 possible keys (excluding a shift of 0).

Cryptanalysis

Brute Force Attack: Since there are only a limited number of possible shifts (25 in the case of the English alphabet), an attacker can try all possible shifts until an intelligible text is produced.

Frequency Analysis: Since the cipher does not alter the frequency of individual letters, an analysis of letter frequencies in the ciphertext can be a useful tool in decrypting it. For example, in English, 'E' is the most common letter, so a high frequency of a particular letter in the ciphertext might indicate it represents 'E'.

Setting Up the Python Script

1. Open your Python environment and start a new Python script.
2. Define two functions, encrypt and decrypt.
3. Ask the user to input a message and a shift value.
4. Encrypt the message using the encrypt function.
5. Decrypt the message using the decrypt function.
6. Print out the original, encrypted, and decrypted messages to see if the decryption works correctly.

Sample Python Code for Testing

```
message = input("Enter a message to encrypt: ")

shift = int(input("Enter shift value: "))

encrypted_message = encrypt(message, shift)

decrypted_message = decrypt(encrypted_message, shift)

print("Original Message:", message)

print("Encrypted Message:", encrypted_message)

print("Decrypted Message:", decrypted_message)
```

Testing

1. Run the script with different messages and shift values.
2. Analyze how changing the shift value affects the encrypted message.

3. Modify your python code such that the user can have 2 choices: either enter an original message with a shift key to be encrypted and decrypted as before, or to enter an encrypted message with a shift key to be decrypted. Run an encrypted message with random shift keys and see how many tries it took to correctly decrypt the message.

Deliverables

Prepare a simple pdf lab report containing your sample message, your encryption and decryption algorithms, as well as the results of the testing section above. Include the names of all group members THAT PARTICIPATED in this lab activity.