Katie Inman

CIS156

Prof. Ling

1 February 2024

Problem Set 1: Encryption

Code Reflection

This code implements a variation of the Caesar Cipher encryption and decryption process. It first loads a list of valid words from a file which is used to check the validity of words in the encrypted message. The Message class is used to represent a message, which generated a Caesar Cipher encryption dictionary based on a given shift and to encrypt the message. The PlaintetMessage class extends Message and adds functionality to encrypt the message using the specified shift. The CiphertextMessage class handles decryption by trying all possible shifts and selecting the one that results in the most valid words, helping to determine the correct shift.

I found that the most difficult part of this assignment was understanding the logic behind what was happening. I referenced the Zybook examples and lessons frequently, especially when it came to properly using classes. It took me a few tries to make sure that the code was functional in the end, but looking at each section individually before I moved on in the assignment.

Pseudocode

Open file “words.txt” for reading

READ the first line of the file

Split the line into words and store in word\_list

PRINT number of words loaded

Close file

Create Message object with input text

Load valid words from ‘words.txt’ using load\_words

PRINT message\_text

Create shift dictionary using build\_shift\_dict with given shift value

Apply the shift to the message using apply\_shift

Create PlaintextMessage object with input text and shift value

Initialize the Message class with the input text

Store the shift value

Build encryption dictionary using build\_shift\_dict

Encrypt message using apply\_shift and store the encrypted message

Get encrypted message text from PlaintextMessage object

PRINT encrypted message text

Create CiphertextMessage object with encrypted message text

Decrypt the encrypted message using decrypt\_message method

Try every possible shift (from 0 to 25)

Apply the reverse shifts and check the number of valid words

Track the shift with the most valid words

PRINT the best shift value and the decrypted message text