

14.07 Assignment Instructions

Instructions: Write a program to perform a frequency analysis of the letters in a message and use the results to decipher a secret message.

Break this project down into sections. Design a plan and then execute it. First, concentrate on just the frequency analyses; you need the results to do the rest of the assignment.

Part 1: Frequency Analysis Lite

1. Create a new project called 14.07 Frequency Analysis in the Mod14 Assignment Folder.
2. Create a client tester class and an implementation class for the frequency analysis in the newly-created project folder.
3. You will need a plaintext document of approximately 500 words or more for this assignment. The `plainText.txt` file downloaded previously is a good example, but you may choose to use a paper or essay you have written for another course, or you may copy and paste something from a website. The longer the better so the frequency distribution of the letters will be more accurate.
4. Now create a subset of the plaintext file to use in the early stages of testing your program. To verify your program is working, use a subset of the document that is small enough for you to manually count the letters. Simply copy and paste the first 15–20 words from your plaintext into a text editor and save the subset of the document as `subset.txt`. Count how many times each letter occurs in the `subset.txt` file and calculate the percentages with a calculator. You will use this information to verify that your program works correctly.
5. Write a program to read from a text file and create a frequency analysis for each letter of the alphabet. Count how many times each letter occurs within the text. Ignore symbols and whitespace.
6. The program should ask the user for the name of the input file. This will allow your program to be reusable with any text file. The output file created should use the same filename with "Freq" added to the end.
7. Run and test the program. Read the `subset.txt` file and determine the frequency of the letters in this simple plaintext document.
8. Print the results neatly to the screen (e.g., control horizontal spacing, alignment, and decimal places).
9. Write the results back to a separate text file called `subsetFreq.txt` that shows the letter, the number of times the letter was found, and the percent frequency of the letter. Each row in the file should represent a different letter.

Part 2: Frequency Analysis of Plaintext

1. Once your program works with the `subset.txt` file, it should work for a text file of any size.
2. Now run your program and, when prompted, ask it to read the `plainText.txt` file or a text file of your choice.

3. When you run the program, the frequency analysis of the letters in the *plainText.txt* file should be printed neatly to the screen, as well as the text file *plainTextFreq.txt*.

Part 3: Frequency Analysis of the Ciphertext

1. Repeat running your program, and this time use it to analyze the encrypted *cipherText.txt* document.
2. The input and output files should be named *cipherText.txt* and *cipherTextFreq.txt*, respectively.

Part 4: Decipher the Message

1. Now that you have the results of the frequency analysis of the plaintext and the ciphertext, place them side-by-side. This is your decryption key.
2. Using the encrypted message in the *secretMessage.txt* file, begin the deciphering process. You do not need to write a program for this. Instead decipher by hand using paper and pencil or a word processing document.
3. The easiest way to decipher the message is to start with small one- or two-letter words. There are very few possibilities for one- and two-letter word combinations, so it will quickly be obvious if you are on the right track.
4. Pick a letter in an encrypted word and find the letter in the ciphertext frequency.
5. Look for the same or a similar frequency of a letter in the plaintext frequency analysis. You may find one or more exact matches or close matches with plaintext letters. Try substituting a corresponding plaintext letter for a ciphertext letter in the secret message. Keep in mind that your first choice(s) may not be correct. This is where the fun begins!
6. Use context to help you decide if you have made a correct choice. For example, if you are looking at a one-letter word and the percentages tell you that "C" matches up with "b," you probably are on the wrong track. But, if "C" matches up with the letter "a" or the letter "i," you can have more confidence that you are making progress. Try substituting "a" or "i" in other words to see if the message starts to unfold.
7. Deciphering the secret message will take some trial and error, so be patient.
8. Continue substituting corresponding letters until the secret message is revealed.
9. In a word processing document, record the deciphered message.

Part 5: Evaluate the Process

1. What difficulties did you run into writing the frequency analysis program? How did you resolve these problems?
 2. How closely did the frequencies of letters in the plaintext and the ciphertext correspond? What could be done to improve the accuracy of the correspondence?
 3. This assignment involved writing a program and then interpreting the results the old fashioned way. Propose an extension to this project that would allow the computer to do all the work so that you only have to evaluate whether the decoded message makes sense. You don't need to actually write another program to do this, just consider how it could be done and clearly describe your solution in a well-written paragraph.
-

