Final Project Description

CSCI204 will contain one final project. This final project will be around the concept of text analysis. Text analysis is a wide area that includes data mining and cryptanalysis. Text analysis is a huge business and used by companies such as eBay, Facebook, Microsoft, and Google.

In particular, this project will focus on four parts: good programming, data organization, data parsing, and data analysis. To insure that the final project goes smoothly for students, the final project will be broken into five checkpoint (Sept 16, Oct 2, Oct 17, Oct, 31, Nov 18) and the final turn in on Dec 5. The five checkpoints will account for 33% of the project grade.

Check Points:

Check points are due the days (Sept 16, Oct 2, Oct 17, Oct, 31, Nov 18) . You will be required to submit these checkpoints in either two ways. The first and recommended way is via git repo link. More about this is given later in this document. The second is via a zipped file with your name and checkpoint number via Moodle. This must include a README and all the python files used in your program (not just the ones modified by that checkpoint). These check points will be graded in the following manner. First, the project will be attempted to run. Second, sample inputs will be tried from the menu. If any these input fail, this counts as missing one step of the check point. Lastly, the code requirement of each step will be hand checked. If any parts of these requirements are not fulfilled, this will count as a missed step. Based on the number of steps that based a score will be assigned to the checkpoint. A 2 will be assigned if the student fulfills over 85% of the requirements. A 1 will be assigned if less than 85% percent but more than 50% is fulfilled. A 0 will be assigned if less than 50% is fulfilled. These can be considered A, C, and F respectfully. All three check points will account for a total of 33% of the total final project grade.

Code Comments:

Though code comments are not checked in the check point, code comments will be checked in the final project. They will account for 10% of the final project grade. The following will be checked for comments.

1. At the beginning of each py, include your name, the name of the function/classes in the file, and a brief overview of what they do using docstring.
2. Before every function, you must state the purpose of the function and any assumptions the function makes (A good way to outline these assumptions is with pre/post conditions).

Final Project Write-up:

The final project requires a write-up that will be turned in with the project. This write-up will have two sections. Section 1 will outline a brief user manual of your project. This should give general directions

to the user to clarify how to use the program. Section 2 will be a reflective piece about your own programming experience. The reflection should outline what you learned, what you struggled with, and how you overcame these troubles. 20% correct spelling and punctuation. 80% fulfill both sections.

Correct File Management:

10% of the final grade will be based on correct file management. If no file management is used (all in one file), the instructor holds the right to give the student a 50% on the whole project. The key ideas examined will be the use of breaking down the code into modulus that make sense and make the code easy to read.

Correct Class, Function, and Variable Use:

This section will grade multiple ideas. This includes but not limited to correct variable names (ones outlined in check points), correct variable style (e.g., class starting with capital, simple variable starting with lower case letter, full words and no single character names for key variables). This includes the correct use of setters and getters for class. Additionally, name functions that need to be included.

Correct Execution:

Correct execution includes multiple topics. First and foremost, correct execution includes that your code will run with cpython version 3.4+ . If you do not know that your code will run under these standards, please see the instructor during office hours. The second topic will be that all functions execute as they should as outlined in guidelines. Though super-efficient code may not be needed except where an algorithm is outlined, all code must be able to run on my system in a reasonable amount of time.

Area	Description	Percent
Check Points	Check points are graded in three categories (2,1,0). See Checkpoint section for more details.	33
Code Comments	Code Comments should be constructive. See comment section	7
Final Project Write-up	One-Three page document outlining how to use your program and what you have learned.	8
Correct File Management	Each class has py	7
Correct Class, Function, and Variable Use		15
Correct Execution	Must run and work as outlined	30
		100

Git Repo:

Though not required. I highly suggest the use of a git repo for this project. This repo allows many nice features. One nice feature is that it keeps a history of changes in case you make a change that you don't know how to fix. It is also a good backup. Moreover, it allows you to work with others on your code at the same time and provides a display of projects you have done that you can share with future employers. The best free public version is Git Hub. Though git is traditionally a command line tool, git hub also provides a GUI via web or executable that will allow you to manage your git repo. The instructor would be more than happy to help any student get started using a git repo.

Final Project Outline Goals:

1. Read multiple files and store in Document Class.
2. Sort by frequency common words and characters of Documents.
3. Use word and character to decode a Caesar cipher.
4. Use word and character to decode a Vigenere cipher.
5. Encode a document with both Casar and Vigenere cipher.
6. Encode/Decode based on a given map.
7. Compare multiple documents based on word and character frequency.
8. Compare multiple documents based on sentence length.
9. Plot distance with a custom 2d plot function.
10. Plot frequencies with a custom histogram function.
11. Write out multiple files from Document class.

Menus:

A full list menus used to access the Final Project Goals will be published before the release of checkpoint 4

Extra Credit:

The instructor may assign up to 10% extra credit to outstanding projects. All projects must meet the project outlined goals to be considered for extra credit. Topics that may receive extra credit are improved plots and detailed plots. Unique style menus (They still must follow the menu outline). The addition of a graphical interface (console interface must still be present). Threaded support.

Some References:

http://en.wikipedia.org/wiki/Caesar_cipher

http://en.wikipedia.org/wiki/Vigen%C3%A8re_cipher

http://www.mathcs.duq.edu/~juola/papers.d/fnt-aa.pdf

http://evllabs.com/jgaap/w/index.php/Main_Page