Marc Deaso

CIS 173 Final Project

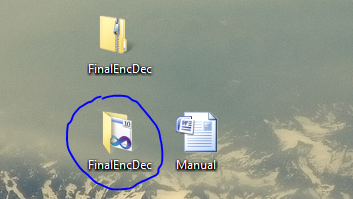
12/3/14

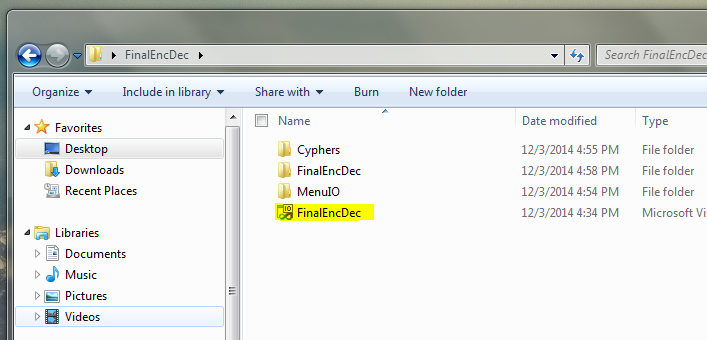
Encryption/Decryption Tool

This guide will talk about and explain how to use the included tool. The application will be able to encrypt text, as well as allow the user to decrypt text easier. Text can be input either directly from the keyboard or from a text file in either case. The program also implements Pearson’s correlation constant to assist in manual decryption.

# Building and running the program

In order to use the tool, we obviously have to run it. We can do this easily by first unzipping the solution to a desired folder, then opening the solution in Visual Studio



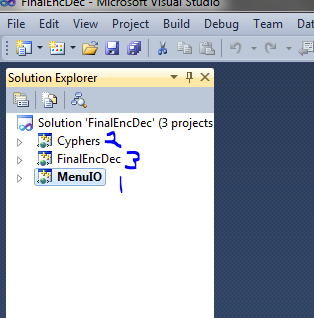


Once the solution is open we can then right click on the individual projects and select build, in the corrrect order to build the solution. The order to build the projects is this:

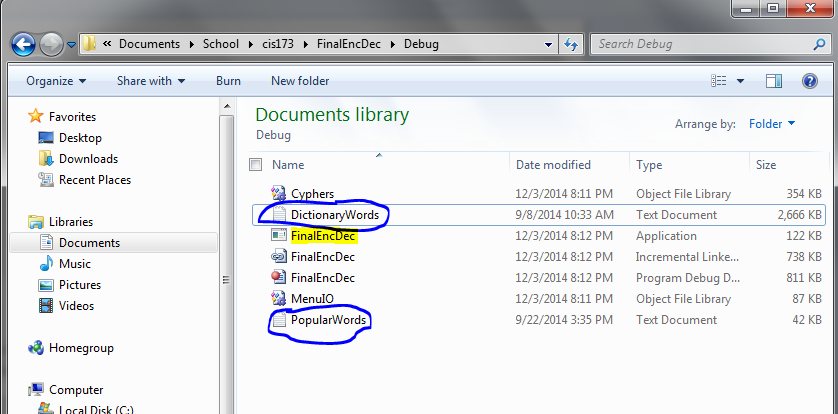
1) MenuIO

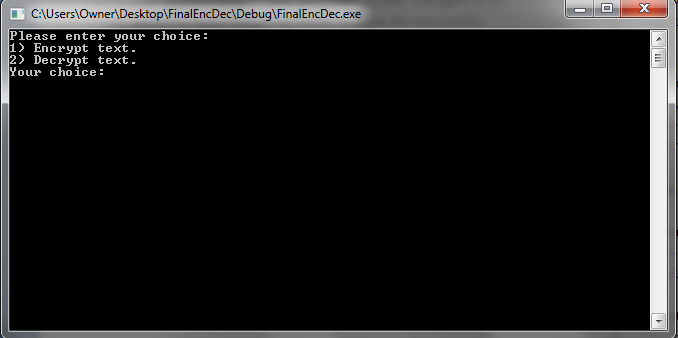
2) Cyphers

3) FinalEncDec



The two dictionary files from the FinalEncDec\FinalEncDec folder must be copied to the FinalEncDec\Debug folder. The solution is now built and ready to run. Navigate to the folder where you extracted the projects, enter the debug folder and run the executable.



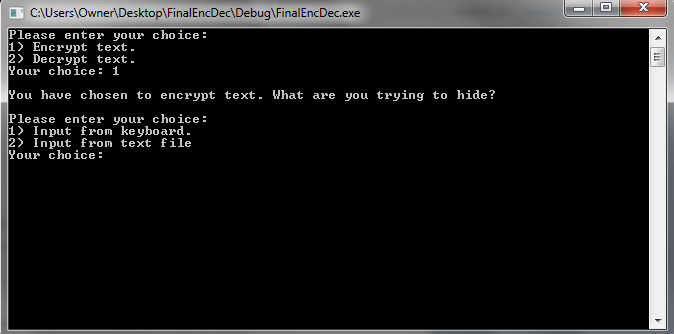


# Using the program

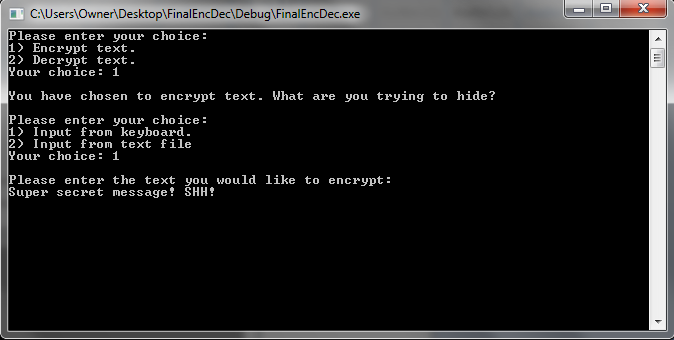
Now you’ve got the program running. How do you use it? The application uses a user-friendly menu system to navigate through its different abilities. To use the menus, just type the number of your choice and hit enter.

## Encrypting

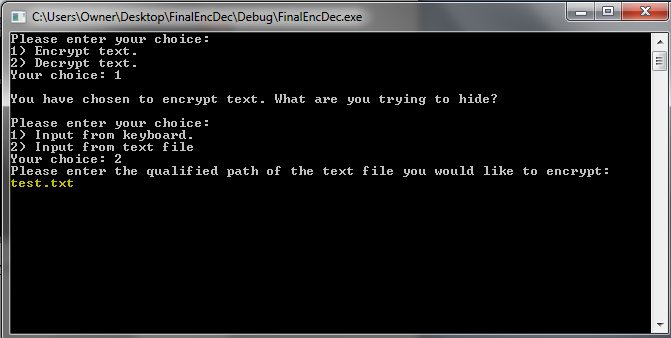
If you have some text that you don’t want others to be able to read, you can use the encryption tool of this program. Choose option 1 on the main screen.



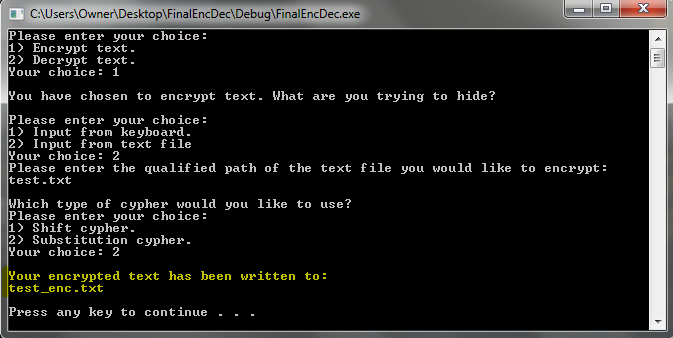
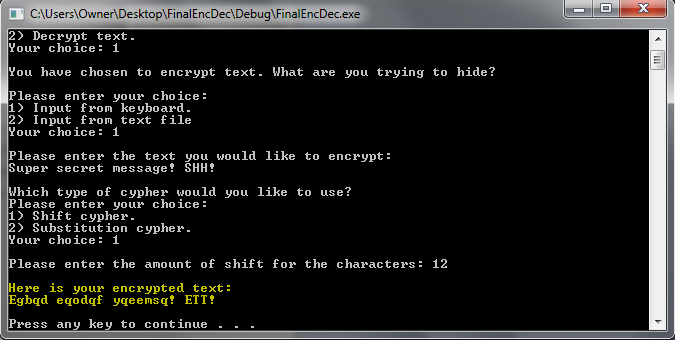
If you choose to enter the text directly from the keyboard, choose option 1 and enter your message.



If you are using a text file choose option 2 and you will be prompted for the file to be read into the program. You can enter the file name only if the file is located adjacent to your executable, otherwise a path is required.



Now the program has your message, you will be asked how to encrypt it. Shifting will change every letter by the same amount. For example, A’s will turn to C’s with a shift of 2. Substitutions are predetermined and swap each letter for a different one in no particular order. After you enter a shift amount or choose a substitution, the program will output your encrypted text to the screen if you entered from the keyboard, or to a file in the same location as the input file but with “\_enc” added to the end.

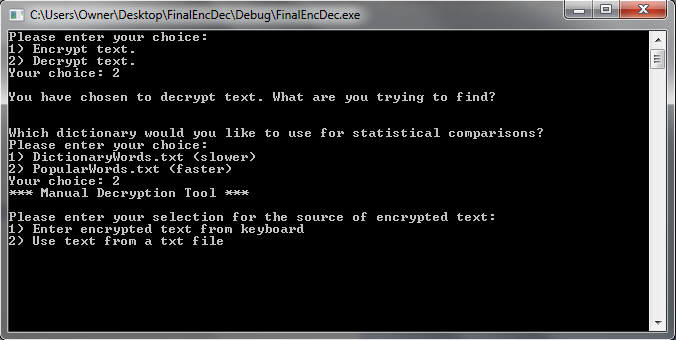


## Decrypting

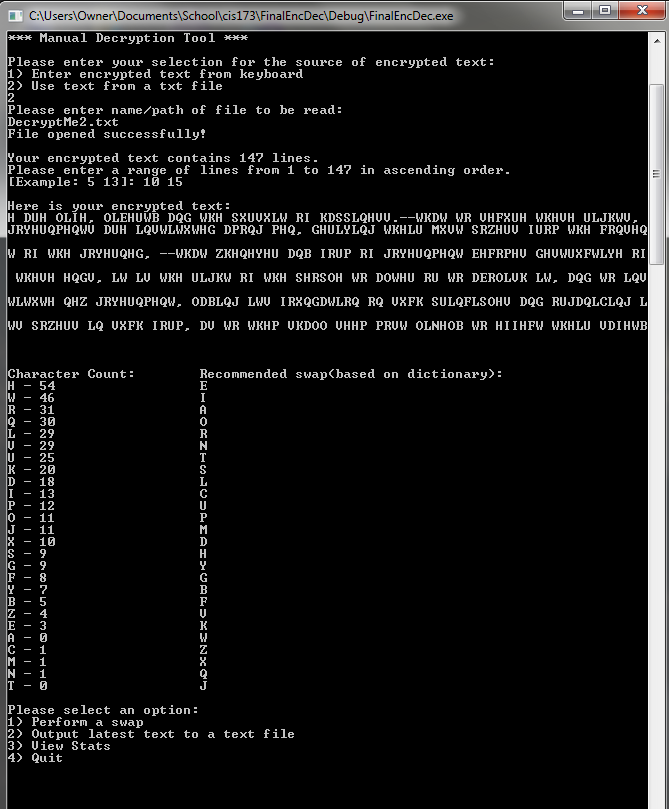
The decryption tool in this program is an assisted manual tool, unfortunately it will not automatically decrypt text that it is sent.

After choosing decrypt from the main menu, you will be asked which dictionary file you want to use. The dictionary files are used to compare character counts to those of your encrypted text making it easier to decide which letters to swap. DictionaryWords is a larger file and may take longer to calculate stats than the smaller PopularWords. Keep in mind that one might be better to use than the other depending on what you are decrypting.

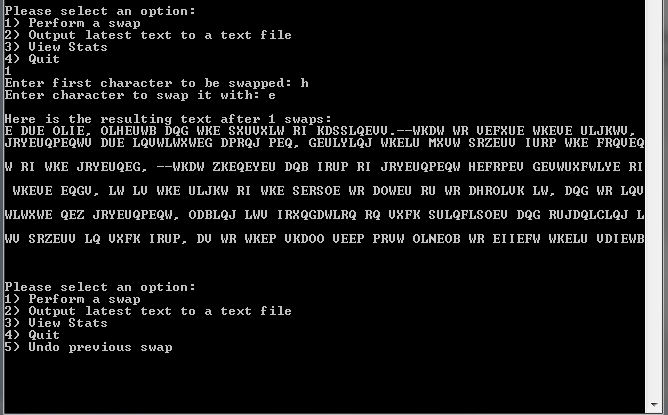
Once you choose a dictionary file (both will take you through the same process with different stats), you are again prompted for either entering text from the keyboard, or from a text file. The process for entering text is very similar to that of the previous section.



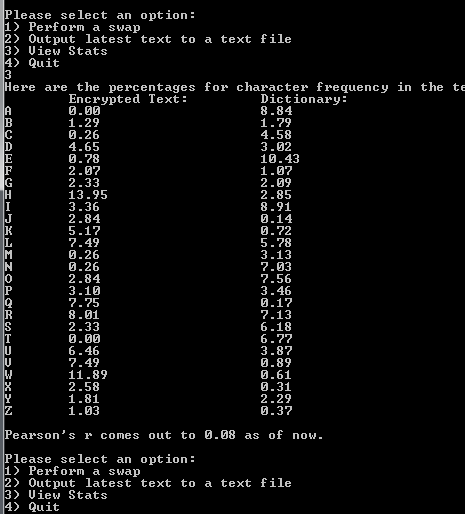
If the text you entered is longer than 20 lines in a text file, you can choose a range of lines to display as the text that you will decrypt and later apply the changes to the entire document.



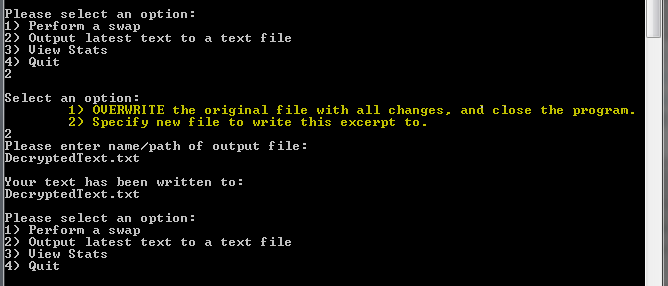
Now the program knows the text that you are trying to decrypt and will give you recommended letter swaps that you can choose to follow. This is the main decryption tool menu. It allows you to 1) make a swap by entering two different letters, 2) output your text with the updated swaps to a text file, 3) view stats like the character counts again and Pearson’s r, 4) quit the program, or 5) undo a swap if one has been made.



As you can see, the fifth option to undo a swap appears after a swap has been made. The updated text is displayed in the console after a swap. Here is an example of the stats page, this can further assist in decryption by observing the r value after swaps. The closer r is to 1, the closer the letters line up with respect to their frequencies.



On the “write to file” page, you can either choose to overwrite the entire original document with the swaps, or choose a new file to write the decrypted excerpt to.



When you are satisfied with the actions you’ve made, option 4 will end the program.