Project 1 Documentation

Mackenzie Zappe

Lab Section 1105

The purpose of the program is to take in various names from a file and sort them by different standards. There can be no more than eight letters in a name and no more than ten names in a file. Once the program copies over the list of names, different functions will sort the names alphabetically and by length. The final task for the program is to print the data to the terminal and to an output file.

The design of my code is based around modularity. I first took the purpose of the program and then divided it into different objectives. The first objective was to read in the list of names. I then decided that this objective had only one step and therefore only one function needed to accomplish it, so I implemented a readFile() function. The next objective was to print the the list to the terminal which again only needed one function to finish, so I implemented a printToTerminal() function. The next objective was to sort the names alphabetically. For this module, I decided I needed to break it down further to accommodate its complexity. The sub-objectives included a swapping mechanism and a sorting mechanism. The swapping mechanism yielded the strcpy(), strcmpr(), and strlen() functions. The sorting mechanism yielded a selection sort function, combining the swapping mechanism with alphabetic specific ordering. Next I had the length sort objective. Here, I employed the previously created swapping mechanism and its functions to combine with a similarly formatted selection sort function, only this time based on length. The last objective for the program was to print to an output file. Like the printing to terminal objective, this is only one step and furthermore the same structure.

My biggest problem was adjusting to the syntax of C++ and its different structuring of the I/O stream. This slowed me down as I had to check the documentation over and over to ensure I was implementing correctly. Another big problem I had was with the strcpy() function. For some reason it kept adding random additional letters to the ends of the names which threw off the further sorting to be done. Eventually I trial-and-errored my way to a solution, but I still do not know why it was messing up in the first place. Design wise I had no major problems. The process of the selection sort was a little tedious to work out but overall made my code easier to implement.

If I had more time I would try and figure out why the strcpy() function wasn’t working with my first implementations. Also I would look for minor syntax touchups to make my overall code more effective.