February 28, 2020 Braelyn Rotman
1006740

Assignment 2 Reflection Report

My Approach To Designing The Program:

1. First I wrote the subroutines randomInt, isWord, and getFilename since they were each short and not dependent on other routines.
2. To start processing the file in processText I iterated through each line and created/printed a substring every time I reached a non alphanumeric character. Each non alphanumeric character was then printed after the substring.
3. For each substring that was created I then checked the length followed by calling is word to see if the substring needed to be scrambled. I decided to use a conditional statement so that non words or short words could be printed and everything else would be sent to the scrambleWord subroutine.
4. By this point I had the overall structure of the program written and I now designed my scrambleWord routine. The main idea for my scrambleWord function was to create a new variable to store the scrambled word and replace the inner characters with asterisks until each character was replaced. Then using a random number from randomInt I could check if the index had already been used by looking for an asterisk and create a different random number.

Was Ada Well Suited to Solving the Problem?

I thought Ada was a good choice in language and was well suited for solving this problem and was significantly easier to use than a language like C. The reason for this is that Ada allows you to easily handle strings. With Ada you can directly access strings at any index and replace any character. The slice function was also vital in tokenizing the lines from files and separating each word. This problem was entirely string based and would have been very difficult in a language such as C since we would have had to use many pointers to create and iterate through strings.

What Particular Structures Made Ada a Good Choice?

As mentioned above the string type, particularly the unbounded string was very useful for the project and made Ada a good choice in language for the problem. The string structure made writing the program quicker and more readable than if pointers and arrays of characters had to be manually created. Ada also provides many helpful string attributes that could be utilized in this assignment. I used the attribute length often. Since I used unbounded strings, I needed the length to know when to end each of my iterations.

Benefits / Limitations?

One of the benefits of Ada that I found is that it allows you to define custom ranges of indexes making programs much more readable. Ada also has the benefit of many packages for things such as string and character handling. These packages help keep programs concise, clean, and readable. Another thing benefit I noticed during this project is that Ada allows for easy memory management. With unbounded strings I didn’t have to worry about overflow and allocating memory. Ada is strongly typed which helps prevent a lot of programmer mistakes. I did not come across any limitations during this assignment and I felt that Ada is a very useful language for projects like this.