Thomas Wiles

Programming Assignment 4

Spellchecking with a Dictionary

March 24, 2017

Abstract: The goal is to spellcheck a text for grammatical accuracy. It is accomplished using a dictionary formed from an array of LinkedLists using a text file dictionary.txt. Once the dictionary.txt file is read, each word can be loaded into the appropriate LinkedList using the first letter of each word. While it might be to our advantage to organize further alphabetically, using the first letter alone proves sufficient. Then, the text file oliver.txt must be spellchecked against the words in the dictionary. This is simple enough once a decent method for splitting the String has been developed. After each line is split into words, they can be compared to the elements of the LinkedList dictionary. Again, this uses the first letter of each word to determine which list is used for comparison. The data format lends itself to this type of comparison. It is easy enough to parse String. Then the word comparison is achieved using the method contains(). The only challenges are that certain characters make the split inconvenient, such as [and], because the method split() does not work well with those. The results are that 866,748 words were found in the dictionary. It had to do an average of 3,564 comparisons to find them. The words not found in the dictionary numbered 40,894, and it had to do an average of 8,724 comparisons to determine this. This means that the average amount of words per letter of the alphabet is 8,724 for the current dictionary.

```
run:
The number of words found is: 866748
The number of comparisons per word found is: 3563.82
The number of words not found is: 40894
The number of comparisons per word not found is: 8724.00
BUILD SUCCESSFUL (total time: 22 seconds)
```