Jonathan Steininger
Programming Assignment 4
Assignment4
3/25/2017

PROBLEM STATEMENT

The problem was to create a rudimentary spell-checking program and to find the average number of comparisons it took to find words in the dictionary and the average number of comparisons it performed when it could not find words in the dictionary.

METHODOLOGY

This spell-checking program was implemented by first creating a searchable data structure that could store the words from a simple dictionary and then to read the text to be spell-checked, extracting only the words and not the numbers or other special characters, and then comparing those extracted words with the words in the dictionary. Whenever a word is either found or not found, values are stored for the total number of times a word is found and the total number of comparisons it took to find those words, or the total number of times a word is not found and the total number of comparisons it took to not find those words. These numbers can then be used to determine the aforementioned averages.

OBSERVATIONS

The results of this program are rather interesting. Because the process for not finding a word in the dictionary for a word from the Oliver text means checking against every single word in the dictionary that begins with the same letter, this means that the average number of comparisons for words not found is simply the average number of words in each linked list that starts with the same letter. Also, because the words in each list are not in alphabetical order, but instead random, the average number of comparisons it takes to find a word in the list should be roughly half because the words may be found equally like near the beginning of the list as at the end of it. The random ordering of the lists, although not necessarily important for that fact, but it is a nice safeguard since it is possible that the words in the text may be more likely to be xa, xb, xc, or xm etc. as opposed to xq, xv, xx, or xz etc. (or vice versa). Nevertheless, this program is not perfect since it cannot properly deal with words containing apostrophes, specifically contractions. Currently, it takes words such as: well-respected, king's, can't, and didn't and then turns them into the words: well, respected, king, s, can, t, didn, and t. This is effective for determining that the words that make up “well-respected” are spelled correctly and that “king” is spelled correctly when it would otherwise mark “king's” as being incorrect, but it artificially increases the number of incorrect words by listing “s” as a word, and it also means that “didn't” is counted as incorrect twice, even though “didn't” is spelled correctly, because neither “didn” nor “t” are in the dictionary. Finding a way to appropriately deal with apostrophes in possessive words and contractions is proving to be elusive.

Results

Words Found:	940258
Words Not Found:	59283
Comparisons Found:	3345813646
Comparisons Not Found:	468560620
Average Comparisons for Words Found:	3558.40
Average Comparisons for Words Not Found:	7903.79