Autocomplete

CSS 143

Author: Matthew Blaisdell

After thinking about this problem, I decided that having all the word in a sorted list was the best place to start. Thus, I have a constructor that adds the words to a list and then uses collections.sort() to put them in alphabetical order.

Now that the words are alphabetized, I thought about how to search for a partial string in the list of words. I decided to use a binary search, of sorts, to find the index of any word that contained the search string. I did this by using the string.contains() method. Then by usingin string.compare(), I was able to determine which half of the words array the search term is included in.

Once I found an instance of the search string in the words array, I then searched the words to the left and right of the found term to find the bounds of the words array that contained the search term, again using the string.contains() method. Once fining the upper and lower bounds, I returned them to the autocomplete method.

The autocomplete method, now having the upper and lower bounds that contain the search term, uses a loop to add all the words within the bounds to an array that is passed back to the test method.

This method is the fastest I could think of. A binary search has a complexity of O(logn), so finding a term should be relatively quick. Depending on how many words there are that contain the search term, the complexity will be increased due to finding the bounds. I still figured this was a relatively quick way to search, given that looking at each individual element would be a complexity of O(n).