Assignment #5 - Text Analysis Algorithms and Datastructure

Marjahan Begum and Anders Kalhauge

Spring 2017

In groups:

Implement a Trie that works on the English alphabet and ⁄. The trie shall save information about each word in a text with the word and it's number of occurences.

Try the data structure on Shakespeares complete works. Write out all words with their counts in alphabetic order.

Tips:

- use the filter to remove unvanted characters
- convert each word to upper case
- create a class e.g. `WordCounter` that holds the word and the count

If you have time:

Implement the `soundex(String word)` method. The `soundex` method calculates a code that reflects the overall spelling of a word. Slightly misspelled words will have the same soundex code as the correctly spelled word.

You can find a desccription of the algorithm here:

`https://en.wikipedia.org/wiki/Soundex`

Create another Trie, that takes a soundex as key and saves all occurences of words with the same soundex. The data class would have the soundex key and a list of words.

Create a small application, that takes a word, misspelled or not, and compares it to correct Shakespeare candidates.

Defend the choice of datastructure with regard of time and space complexity (big O).

The solution accompanied with a description in a README.md file should be uploaded (pushed) to a git repository, and a link should be handed in, in Moodle no later than Wednesday May 17. 12:00.