

This assignment was locked Apr 22 at 11:59pm.

1. Write a Ternary Search Trie that takes a command-line argument specifying the name of a file containing a dictionary of English language words. Then insert these words into a Ternary Search Trie. After that, read in a regular expression pattern from standard input and print out all of the words in the trie that match this regular expression. Continue reading until end of file: cntr-d. Use the Ternary Search Trie code provided on the book's web site. Use the Java's Matcher and Pattern class and their methods to handle the regular expressions.
2. Then replace the Ternary Search Trie with a binary search tree.
3. Collect data about the time for a successful search and the time for unsuccessful searches for the Ternary Search Trie implementation and for the Binary Search Tree implementation. You will need to think about how to do a fair comparison. You will probably want to gather some data about searching with various types of regular expressions. Also, you may want to choose words at random for which to search. That is, if you search for a word that is shorter it may be found more quickly in a Trie than in a BST, or vice versa. Or words that come earlier in the dictionary may be different from those later in the dictionary (or maybe not). Or perhaps, the order in which you insert them into the tree makes a difference. Think about how to do a fair comparison. I suspect that choosing words at random may be the best.
4. Write an analysis of the data that you collected. Explain any differences. For instance, is the Ternary Search Trie better because we are searching for strings since that is its purpose.

Here is a file of words, one per line, to use for testing. This is from <https://github.com/dwyl/english-words/blob/master/words.zip> : [words.zip](#)

Include JUnit tests and evidence of coverage.

If you do not know what a regular expression is, see section 5.4 of the textbook, or this or a similar web site:

- <https://medium.com/factory-mind/regex-tutorial-a-simple-cheatsheet-by-examples-649dc1c3f285>