

Objectives

1. Using the *ArrayList* class.
2. Implementing and using classes

Overview

A *concordance* is defined as a table containing each *word* that occurs in a text file, together with a count of the number of times each word occurs (its *frequency*), and a list of the *line-numbers* where the *word* occurs. For example:

<u>Text File</u>	<u>Concordance</u>
See Spot run.	bite 1 [3]
See Jane run.	jane 2 [2, 4]
See Spot bite.	run 4 [1, 2, 4]
Run Jane, Run!	see 3 [1, 2, 3]
	spot 2 [1, 3]

A *word* is any sequence of any non-white-space characters, beginning and ending with non-punctuation symbols. The following are punctuation symbols: . , : ; ? ! () ‘ “.

Specific Requirements

1. Implement a class, *WordRecord*, to represent the information about a *word*. The class must have instance variables for a *word*, the *frequency* of that word, and its *line-numbers*. In the example above, each row of the *concordance* is a *WordRecord*. **You must use an *ArrayList*, to store the *line-numbers* in ascending order.** There must be no duplicates! A public interface is specified in the *WordRecord* start-up file.
2. Implement a class, *Concordance*, to represent a *concordance*. The implementation must be as an *ArrayList* of *WordRecord* elements. A public interface is specified in the *Concordance* start-up file. The *Concordance* constructor scans a *text file* to extract individual *words*. As each *word* is extracted, it is “pruned” of leading and trailing white-space and punctuation. The *Concordance table* is **updated via a helper mutator**: either a new *WordRecord* is **inserted** into the *Concordance*, or an existing *WordRecord* is updated. There are no public mutators.

The *Concordance* methods include accessors, queries about the *text file*, and *word* queries. **The word queries must all be implemented via a binary-search helper.**

3. A client/test program, and some test data files, are provided. Run and debug your *Concordance* implementation on at least these data files.

Submitting your Assignment

Upload your source (**.java**) files in SCIS Moodle by the due date. **No late submissions.**