Assignment 2 – Selection in a Collection

Assigned: Wednesday January 23, 2013
Due: Wednesday January 30, 2013 11:59 PM
Assignment Type: Individual

Problem Description

This assignment focuses on implementing the methods of a class much like java.util.Collections – a library class with static methods designed to provide useful functionality on general Collections of elements. Unlike in the previous assignment, generalized programming is not only encouraged but required. You must correctly implement all method bodies of the provided Selector class. Your implementation must adhere exactly to the API of the Selector class, as described in the source code comments. Deviation from any aspect of the API, or the items listed below, will result in a significant deduction of points. Other specific requirements and notes:

- You must not add any public methods to the Selector class. You are free to add any private methods that you think appropriate, but you can't add any public methods.
- You must not add any fields, either public or private, to the Selector class.
- You must not modify the constructor of the Selector class in any way. The constructor was intentionally made private so that the class cannot be instantiated.
- You must not add any other constructors to the Selector class.
- You must not put your source code in a Java package.
- You must not add any import statements to those already in the Selector class.
- You may not change any import statement in the Selector class.
- None of the methods in the Selector class can modify the Collection parameter in any way. More generally, your methods can have no side-effects outside their own scope.
- The methods in the Selector class must only return values. No printing or output of any kind is allowed.
- Sorting can only be used in the kmin and kmax methods. Sorting the Collection parameter or a copy of it in any other method is not allowed. If you choose to sort in kmin and/or kmax, you can do so by calling java.util.Collections.sort(List, Comparator) method. You must use the fully-qualified name (no importing Collections) and you are allowed at most two calls to this method – at most one in kmin and at most one in kmax.
- The kmin and kmax evaluation must be interpreted in terms of a *dense ranking* (http://en.wikipedia.org/wiki/Ranking). For example, suppose the array contained the six elements (five distinct values) [12, 2, 8, 4, 2, 10]. Then for k = 1, kmin = 2 and kmax = 12; k = 2, kmin = 4, kmax = 10; k = 3, kmin = 8, kmax = 8; k = 4, kmin = 10, kmax = 4; k = 5, kmin = 12, kmax = 2.
- The use of the ArrayList class is allowed only in kmin, kmax, and range. No other method is allowed to use an ArrayList or any other implementing class of Collection.
- Your source code must not generate any compilation warnings of any kind.
- You may not use the @SuppressWarnings annotation.
- Record your name and TigerMail in the provided @author tag.
- Replace the date in the @version tag with the current date each time you work on the file.
- Your submission will be graded against a test suite written by the course staff. If your submission does not compile with the test suite, you will receive a score of zero points for the assignment.
- Your submission must be solely your own work. Discussing ideas and general approaches to a problem is fine, but sharing source code, even small sections of it, is considered a violation of the academic honesty policy.

- You are encouraged to ask and answer questions on Piazza regarding the assignment.
- Good coding style is expected. An excellent discussion on style can be found in the CACM article Coding Guidelines: Finding the Art in the Science by Green and Ledgard (http://cacm.acm.org/magazines/2011/12/142527-coding-guidelines-finding-the-art-in-the-science/fulltext). Run Checkstyle with the configuration file for 2210 on your Selector.java file before submitting your solution. Try to only submit code that passes the Checkstyle audit.

Assignment Submission

To submit your solution, you must only turn in Selector.java. You must upload this single file to Canvas no later than the date and time indicated. If you do not turn in Selector.java, you will receive a score of zero points for the assignment. If you turn in more files than Selector.java, the others will be ignored. If your submission does not compile with the assignment test suite, you will receive a score of zero points for the assignment.