09. Ninth Assignment – Sorting

For this assignment you will instrument a generic version of the insertion sort method to illustrate how insertion sort works. The non-generic version of the insertion sort method occurs in Listing 23.1 on pages 863-864 of your textbook. See the video on the course website for an example of instrumenting code. Once you have instrumented the method, create a driver program that initializes three different arrays – one of Integers, one of Strings, and one of FacebookUser objects – and call the insertion sort method on each one. Before you can do this, you will need to make a generic version of the insertion sort method and use it instead of the exact listing on pages 863-864.

Next take a look at pages 871-872 in your textbook, Listing 23.8, which shows an implementation of quick sort that works for an array of int (integers). You will need to make a new version of this with a few changes:

1. Create a generic version of the method that works on all Comparable data types.
2. Choose the pivot based on the median of the first three values. (If there are only one or two values in the array, then just use the first one.)
3. Instrument the method to show the pivot and sub-arrays for each pass.

Finally, shuffle the Integer, String, and FacebookUser arrays you used to test the insertion sort method and use them to test the quick sort method. You can use the shuffle in the Collections class for this, but you will need to figure out how to temporarily convert your array to a list. Alternatively, you can write your own shuffle method.

Making these kinds of small changes to the sorting algorithms will ensure that you have a good understanding of how they work.

You will be graded according to the following rubric (each item is worth one point).

- The generic version of the insertion sort method is correctly implemented
- The insertion sort method is instrumented
- A driver program tests the execution of the insertion sort method on three different types of arrays of: Integers, Strings, and FacebookUser objects.
- The generic version of quick sort is correctly implemented
- The pivot in quick sort is chosen based on the median of the first three values
- The quick sort method is instrumented
- The arrays sorted by the insertion sort method are shuffled before using them to test the quick sort method
- The driver program tests the execution of the quick sort method on three different types of arrays: Integers, Strings, and FacebookUser objects
- The program compiles and runs
- The program is clearly written and uses standard coding conventions

- ***Note:*** *If your program does not compile, you will receive a score of 0 on the entire assignment*
- ***Note:*** *If you program compiles but does not run, you will receive a score of 0 on the entire assignment*
- ***Note:*** *If your Eclipse project is not exported and uploaded to the eLearn drop box correctly, you will receive a score of 0 on the entire assignment*
- ***Note:*** *If you do not submit code that solves the problem for this particular assignment, you will not receive any points for the program's compiling, the program's running, or following standard coding conventions.*