Notes on the sample exams

The two samples exams (from Spring 2000 and 2003) are old, but still quite representative of the types of questions you’ll find on the final this semester. Of course, we’ve changed our curriculum since 2000, and Java itself has evolved, so here are some notes that will help you think of those exams in a modern and relevant way.

**Spring 2000**

Question 1: Assume that the boolean variable `hasAllOnce` is initialized to false.

Question 2: If you do either of these, part b is much more interesting. (Also keep in mind that in discussing recursion, we focused on recursive structures, such as recursive lists. So while this question can be fun, it does not reflect our current focus.)

Question 4: Coming up with the “number of comparisons” for the sorted array is actually quite tricky. (We probably won’t ask anything this tricky – at least not intentionally!) It’s interesting to think about how you would implement the search. That’s more important than coming up with a “number of comparisons”.

Question 7: Change `BufferedWriter` to `PrintWriter`.

Question 8: The two buttons should be declared to be of type `JButton` (not `Button`). This means that all instances of `Button` in the code should be read as `JButton`.

The instance variables should also be declared as `private`.

**Spring 2003**

Question 2: Ignore this question. We no longer address “Concurrency” as a specific topic in 134.

Question 4d: At least in Andrea’s section, in writing SelectionSort, we started by finding the smallest element and swapping it into the front of the array.

Question 8: Ignore this one. It’s a fun question, but it really isn’t the style of question we’ll ask. Besides, we didn’t look in lots of detail at the “broccoli” problem.

To practice a recursion question, see question 5 on the sample midterm we posted in October.