EECS2030 Fall 2018
Advanced OOP
Lab Test 2 Preparation Exercise
Time Limit: 80 minutes

Name: (Last, First) ⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯

Student ID ⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯

1   Programming Exercises

- Download and import this starter project.
- This exercise is about a simple application of 2-D points, each with the x- and y-coordinates. A line contains its start and end points. A line collector stores a collection of points. Points in a point collector are *positioned* according to their chronological order of being added (i.e., first-added point at position 0, second-added point at position 1, and so on). Two points are equal if their coordinate values are qual. Two lines are equal if their start and end points are equal. Two line collectors are equal if they store the same number of points, and points at the corresponding positions are equal.
- As before, you are **only** allowed to use primitive arrays.
- There are two packages (`aggreagation` and `composition`), each of which containing a JUnit test class for you to study and infer the necessary classes and methods.
- Before you start this programming exercise, make sure you understand the difference between aggregation (where sharing and thus aliasing is allowed) and composition (where sharing is forbidden).
- You are advised **not to** dive into coding right away and just focus on passing the assertions. The actual test will be similar but with a different example, so merely passing all tests in the exercise (without understanding how things work together) will not lead you far.
- Instead, first study the tests assertions in the two packages: they are very similar, but the differences indicate the distinct requirements of aggregation and composition (in particular the expected results from using == versus `equals` to compare objects).

2 Written Exercises

1. Consider the following statements:

   (**A**) $3n + 7$ is $O(n \cdot log(n))$
   (**B**) $3n + 7$ is $O(n)$
   (**C**) $3n + 7$ is $O(1)$
   (**D**) $3n + 7$ is $O(2^n)$
   (**E**) $3n + 7$ is $O(log(n))$
   (**F**) $3n + 7$ is $O(n^2)$

   (a) Which of the above statement or statements are *correct*?

   [of 10 marks]

   (b) Among the above statement or statements that are *correct*, which one is the most *accurate*?

   [of 5 marks]

   (c) Justify your answer to the previous question. That is, clearly explain why it is more *accurate* than all other *correct* statements.

   [of 10 marks]

2. In order to prove that $f(n) = 4n^3 - 5n^2 + 59 + n^4 + 9n$ is $O(n^4)$, you need to choose values for two constants: constant c as a factor for n^4 and constant n_0 as some starting value of n.

   (a) Write down the precise condition for which c and n_0 must satisfy in order for the proof to succeed. **Hint:** Your answer should involve n^4, $f(n)$, c, and n_0.

   [of 5 marks]

   (b) Give values of c and n_0 that will complete the proof.

   [of 5 marks]

3. Consider the following Java program:

```
1  duplicatePrint(int[] a, int n)
2    int sum;
3    for (int i = 0; i < n; i++) {
4      for (int j = 0; j < i; j++) {
5        for (int k = 0; k < 5; k++) {
6          System.out.println(a[k]);
7        }
8      }
9    }
```

Determine the **most accurate** asymptotic upper bound of the above program, using the big-Oh notation. You **must** show in detail how you determine the bound. Without a convincing derivation process, you will only receive <u>partial</u> marks.

[of 15 marks]