Unit 4: Arrays and Array List Exam

What the Exam Covers:

- Chapter 12 & 13 in the text book
- All Activities in Unit 4 posted on the course website, including Coding Bat questions, the Chomp project, assignment questions, and all review materials.
- Note that the example questions in this review package are the kinds of questions you can expect on the test.

Topics to Know:

Arrays

- Declare & create arrays using the new keyword and the { } shortcut method.
- Access specific elements in an array using [index].
- Access an array's length.
- Access the last element in an array of length n
- Use a loop to access all elements in an array and do something with them. For example, write a method that returns the average all the values in an int array, or converts all the words in a String array to pig latin.
- Use a loop to access all elements in an array and build a new array from it according to some criteria. For example, write a method that takes an int array as a parameter and returns an int array with all the negative values removed, or write a method that takes an alphabetically sorted String array and one additional String and returns a new array with the additional String inserted in the alphabetically correct position.
- Use a loop to answer a true or false question about the contents of an array. For example, write a method that takes an int array as a parameter and returns true if there are consecutive runs of three or more of the same number, false otherwise. Or, write a method that returns true if an Array of Strings is sorted alphabetically, false otherwise.

2D Arrays

- The chomp project is your guide here, as it uses all the things you need to know about 2D Arrays
- Declare & create 2D arrays using the new keyword and the { } shortcut method.
- Access specific elements in a 2D array using indexs [row][column]
- Compute the number of rows and columns of any given 2D array.
- Use a nested loop to access all of the elements or a specified region of elements in a 2D array, such as a single row / column (like countInRow() and countInColumn()) or a rectangular subsection of the array (like fillRect()).

ArrayLists

- Declare and create an ArrayList
- Know how the ArrayList methods work (p. 366)
- Understand the difference between size and capacity for ArrayLists

- Remove specific elements in an ArrayList using remove(), and understand the pitfalls of removing elements in a loop.
- Review how get(), add() (with one and two parameters) and set() works and be ready to read code that uses them and to use them yourself.
- Review how contains() and indexOf() works and be ready to use them (it's just like Strings).
- Understand the terms "Auto-boxing" and "Auto-unboxing."
- Use wrapper classes to store primitive data types in ArrayLists
- Use a loop to access all elements in an array and do something with them. For example, write a method that takes two lists of Integers of varying sizes, each one sorted from smallest to biggest, and return a new ArrayList that combines the two lists. The new list should also be sorted. Or, write a method that takes lists A and B as parameters and returns a new list with only the values that are in both A and B.