Lab 13
Arrays

Overview:

A *data structure* is a method of storing a collection of related data items in memory. The simplest data structure is an array. Arrays are internal data structures. This means that they are used only during the execution of the program in which they are defined.

An *array* is a sequence of variables that share a common name and are the same data type. These variables are distinguished by their position within the array. This position is indicated by a *subscript* or *index*. In Java, the subscript is enclosed in square brackets such as: month[9].

Advantages of using arrays are:
- Arrays allow us to use one variable name for many items.
- Arrays allow us to use the same code to reference different memory locations.
- Arrays allow us to store data for later use in a program.

Set Up
- Create a new Java project in your Eclipse workspace named: **Lab13**
- In the *src* folder, create a package named: **edu.ilstu**
- Import the following file at *T:IT168/Labs/Lab13* into your project.
    o *ArrayDriver.java*

Create a class called *ArrayUtility.java* which will contain methods to perform array manipulations. There are no instance variables required in this class. Write each of the methods to solve each task as described below. Be sure to include proper Javadoc comment blocks for each of the methods.

You have been provided with a test driver class. All except the first test have been commented out to allow you to test one method at a time as they are developed. As each method is written, uncomment the code to test that method.

Class diagram:

ArrayUtility
+ findPercentageGreaterThanTwenty(intArray:int[]):double + findSmallestInt(intArray:int[]):int + findLargestInt(intArray:int[]):int + findDifferenceLargestSmallest(int[] intArray):int + printReverseOrder(intArray:int[]):void + increaseValues(intArray:int[]):void

Requirements for each method. All of these methods receives an integer array of any length that has all of the elements filled.

1. findPercentageGreaterThanTwenty
   a. Write the code to return the percentage of elements greater than 20
2. findDifferenceLargestSmallest
   a. This method should use the findSmallestInt and findLargestInt methods and then calculate the difference of the smallest from the largest.
3. printReverseOrder
   a. Write the code to print all of the values in the array to the screen in reverse order. Put each number on a separate line and label it with the element name.
   b. Example Output of one element: intArray[9] = 22
4. increaseValues
   a. Write the code to increase the value in each element of the array by a factor of 4.

Submission

Submit the following file to ReggieNet by the beginning of your next lab.
- ArrayUtility.java