CECS 174 – Assignment 12 – Statistical Analysis

A statistics teacher has a class of 50 students and wants to do some statistical calculations on their grades. Write a program that has an array of size 50 to hold the grades. Since it would be too much work to enter all of the grades into the array, populate the array with random values from 0-100.

Create a menu that allows the user to apply these functions to the arrays:

1. Display the grades – display the grades as a single column.
2. Display the sorted grades – display the grades as a sorted list.
3. Calculate the average – display the average of the class.
4. Calculate the median – display the median of the class.
5. Calculate the mode – display the mode of the class.
6. Calculate the standard deviation – display the standard deviation of the class.
7. Minimum / Maximum – display the smallest and largest grades.

Hints:

1. You can use either arrays or ArrayLists, whichever you prefer.
2. Some of the calculations require the array to be sorted. Use any of the algorithms given in Lecture 22 to sort your array.
3. You will need to copy and compare arrays, so the code from Lecture 21 will also be useful.
4. Create copies of the arrays to perform the sort on, that way the user can still view the original array when they choose option 1.
5. Calculating the mode requires that you keep track of the number of times a particular number appears. An easy way to do this is to create an array where the indices are the grades, and the values are the number of times that grade appeared.
6. As always, check all user input for validity. Create methods for each of the functions and any others that you might find to be necessary. Add a menu option to quit the program and allow it to repeat until the user decides to quit.

Extra Credit – 2 points – Find the Ace Simulation

Generate a deck of 52 cards and shuffle them. Then draw from the top of the deck until an ace appears. Record the number of cards that you needed to flip to get to the ace. Run this simulation 10,000 times, calculate and display the average number of cards needed to flip to get an ace. Write methods to initialize the deck, shuffle the deck, display the average and anything else that will help you break up your program into smaller pieces.

Hints - To keep your program simple:

1. Create an array of 52 strings (or ints, or booleans) that allow you to represent just the 4 aces and then everything else.
2. Use a counter to iterate through the deck (array).
3. When shuffling, just pick two random spots in your array and swap their values. Repeat this many times (500 is more than sufficient) to shuffle the array.
4. Make sure that you do not count the Ace when it is finally flipped.