CS 1400 - Assignment #6

Maximum Points: 20 pts

Topics:

- Arrays

Coding Guideline (You will be graded on this)

1) Give identifiers semantic meaning and make them easy to read (examples numStudents, grossPay, etc).
2) Keep identifiers to a reasonably short length.
3) Use uppercase for constants. Use upper camel case for classes. Use lower camel case for all other identifiers (variables, methods, objects).
4) Use tabs or spaces to indent code within blocks (code surrounded by braces). This includes classes, methods, and code associated with ifs, switches and loops. Be consistent with the number of spaces or tabs that you use to indent.
5) Use white space to make your program more readable.
6) Use comments to explain how the parts of your program work.

Important Note: All submitted assignments must begin with a descriptive block comment (multi-line comments) similar to the one shown below. It must contain your name and the other information illustrated. To avoid losing trivial points, make sure this comment header is included in every assignment you submit, and that it is updated accordingly from assignment to assignment.

```
/*----------------------------------------------------------------------------
// AUTHOR: YOUR NAME
// FILENAME: TITLE OF THIS SOURCE FILE
// SPECIFICATION: DESCRIPTION OF THIS PROGRAM
// FOR: CS 1400 - ASSIGNMENT #6
// TIME SPENT: HOW LONG IT TOOK YOU TO FINISH THIS ASSIGNMENT
//*/
```

Program Description

Your assignment is to create a class called Collection in a file called **Collection.java**. (there is no main method in this class). A class Collection has an array of integers and a count (integer) as instance variables. The variable **count** keeps track how many integers are stored in the array. The variable name for the array of integers is **numArray**.

Note: You need to distinguish the array size (capacity) and "count" that keeps track of numbers added to this array so far.

Method	Description of the Method
public Collection(int arraySize)	It constructs an empty Collection object with an array capacity specified by the integer parameter "arraySize".
private int search (int searchingNum)	It returns the index of the number specified by the parameter. If the number is not found, it returns -1. It is a helper method.
public boolean add(int numberToAdd)	The method checks if the integer specified by the parameter exists in the array (this should be done using the search method to see if it returns -1 or not) and also checks if the array has not reached its capacity. If both are satisfied, the number is added to the array at the smallest available index. If the array reached its capacity, double its size by calling the method doubleArray() and add the number. If the number is added successfully, then the method returns true. If the number already exists in the array, the new number will not be added, and the method returns false.
public boolean remove(int numberToRemove)	The method checks if the integer specified by the parameter exists in the array (this should be done using the search method to see if it returns -1 or not) and if it does, it removes the number and then shifts all the integers one position to the left and returns true. Otherwise, it returns false.
private void doubleArray()	It is a helper method that doubles the capacity of the array (numArray).
public int findLargest()	It finds the largest integer and it returns it. See the example on page 277 in your book
public double computeAvg()	This method computes the average of the integers stored in the array
public int countPositives()	This method returns the number of positive integers stored in the array
public String toString( )	Returns a String containing a list of numbers stored in the array. An example of such string can be: {3, 6, -1, 3, 23, -50, 43} The string should start with a '{' and end with a '}'.

The class **must** include the following constructor and methods. (If your class does not contain any of the following methods, points will be deducted.)

Save the Collection class in a file called Collection.java and use the following program stored in Assignment6.java, which has the main method to create new Collection objects and to test your class. **You do NOT need to modify Assignment6.java.** The program will ask a user to enter a size for the array. Then it will show the following menu to a user:

Sample run: user input is in RED

```
Please enter a size for the array.
5

Command Options
-----------------------------------
a: add an integer in the array
b: remove an integer from the array
c: display the array
d: find largest
e: count positive numbers
f: compute average
?: display the menu again
q: quit this program
Please enter a command or type ?
```

a

Please enter an integer to add.
5

5 successfully added.

Please enter a command or type ?
a

Please enter an integer to add.
-10

-10 successfully added.

Please enter a command or type ?
a

Please enter an integer to add.
8

8 successfully added.

Please enter a command or type ?
c
{5, -10, 8}

Please enter a command or type ?
a
Please enter an integer to add.
-4

-4 successfully added.

Please enter a command or type ?
b
Please enter an integer to remove.
8

8 successfully removed.

Please enter a command or type ?
c
{5, -10, -4}

Please enter a command or type ?
b
Please enter an integer to remove.
5

5 successfully removed.

Please enter a command or type ?
c
{-10, -4}

Please enter a command or type ?
a

Please enter an integer to add.
12

12 successfully added.

Please enter a command or type ?
a

Please enter an integer to add.
2

2 successfully added.

Please enter a command or type ?
c
{-10, -4, 12, 2}

Please enter a command or type ?
d
The largest number is: 12

Please enter a command or type ?
f

The average is: 0

Please enter a command or type ?
a

Please enter an integer to add.
23
23 successfully added.

Please enter a command or type ?
a

Please enter an integer to add.
6

6 successfully added.

Please enter a command or type ?
c
{-10, -4, 12, 2, 23, 6}

Please enter a command or type ?
d
The largest number is: 23

Please enter a command or type ?
e

```
The count of positive numbers is: 4

Please enter a command or type ?
g
Invalid input!

Please enter a command or type ?
e

The count of positive numbers is: 4

Please enter a command or type ?
f

The average is: 4.833

Please enter a command or type ?
a

Please enter an integer to add.
3

3 successfully added.

Please enter a command or type ?
a

Please enter an integer to add.
7

7 successfully added.

Please enter a command or type ?
c
{-10, -4, 12, 2, 23, 6, 3, 7}

Please enter a command or type ?
q
Press any key to continue . . .
```

Helpful hints for doing this assignment:

- work on it in steps – write one method, test it with a test driver and make sure it works before going on to the next method
- always make sure your code compiles before you add another method
- your methods should be able to be called in any order

Submit your homework by following the instructions below:

****

- Submit your Assignment6.java and Collection.java file on GradeScope. Your assignment will be graded only if it is submitted there, NOT on Canvas or sent by email.
- Collection.java should have the following in order:
  - In comments, the assignment Header described in "Important Note".
  - The working Java code requested in “Problem Description”.
- The Assignment6.java file must compile and run as you submit it.

Important Note: You may resubmit as many times as you like until the deadline. Only your last submission will be considered.

NO LATE ASSIGNMENTS WILL BE ACCEPTED. ALWAYS SUBMIT WHATEVER YOU HAVE COMPLETED FOR PARTIAL CREDIT BEFORE THE DEADLINE!