CS111 Programming Fundamentals II
Lab 1: CS110 Review, Arrays
Due via Canvas

Computer Science

This first lab is meant to help you brush up on several concepts from CS110. You'll create a class, `ArrayProcessor`, with several methods to manipulate the entries in an array of 3 `int`s. In the course of this lab, you'll utilize the random number generator, and brush up on UML diagrams. You'll write a short program to demonstrate the use of the class `ArrayProcessor`. This lab is being provided to you as a hard copy and in Canvas. All future labs will be available on the course Canvas page only.

I. Accessing your account

1. Log onto any of the lab computers, and login to Canvas using your CWU account name and password. If you have questions please ask.
2. Go to Files->labs->lab1 to access this lab on-line.

II. The `ArrayProcessor` Class

The ArrayProcessor class contains three methods. One prints the contents of an array, another sorts the contents of an array, and the third reverses the contents. The class is meant to process arrays of three `int`s For example, if:

`anArray` =	4	7	1

Then the sorted version of anArray would be:

`sortedArray` =	1	4	7

And reversing the **<u>sorted</u>** array would be:

`reverseSorted` =	7	4	1

Using the IDE of your choice, create the .java class depicted in the below UML diagram. Refer to sections 7.4 and 7.5 of the textbook for details about methods that receive an argument and/or return a reference variable to an array. UML diagrams are explained in section 6.2 of the book. The textbook references are correct for the "watermelon" book. If you have a newer edition you might need to find correct sections in your book to look material up.

`ArrayProcessor`

`+ sort3IntArray(anArray : int[]) : int[]`
`+ reverse3IntArray(anArray : int[]) : int[]`
`+ print3IntArray(anArray : int[]) : void`

Each of the three methods receives as an argument a reference variable to an array of integers. Two of the methods return a reference variable to an array object. The method `sort3IntArray` returns a reference to an `int` array that contains the sorted, from smallest to largest, entries in the receiving `anArray`.

Hint 1: for each of the methods `sort3IntArray` and `reverse3IntArray`, create an int array for use in that method, and populate it with entries that you retrieve from the receiving anArray object.

Hint 2: there are many ways to sort an array, which you'll learn about in CS301 and CS302. However, you don't need a fancy algorithm to sort a small array; it can be done using a brute force approach, especially in this case because the input array that you'll be creating will have only 3 entries. You can enumerate all of the possible combinations of entries in `anArray` using an if-else loop. For example, if the input array has entries 1, 6, 3, then you'd check whether 1 < 6, and then whether 6 < 3, and perform swapping when necessary to rearrange the entries to 1, 3, 6.

II. The ArrayProgram class

Write a class, `ArrayProgram`, with a main routine that has the below pseudocode. You'll need to import the `Random` class and generate numbers with an upper bound (see section 4.11 of the "watermelon" textbook).

```
// create an array, anArray, of 3 ints
// create an instance of the Random class

// generate a random number between 0 and 10, and place it into the 0th index of anArray
// generate a random number between 0 and 10, and place it into the 1st index of anArray
// generate a random number between 0 and 10, and place it into the 2nd index of anArray

// invoke the print3IntArray method of the ArrayProcessor class and print the anArray entries
// invoke sort3IntArray method of the ArrayProcessor class, and print the sorted array
// invoke the reverse3IntArray method of the ArrayProcessor class, and print the reversed array
```

III. Sample Invocation

Three sample invocations of the program are shown below.

```
The original array
 Entry 0 is 8
 Entry 1 is 2
 Entry 2 is 9


The sorted array
 Entry 0 is 2
 Entry 1 is 8
 Entry 2 is 9


The sorted reversed array
 Entry 0 is 9
 Entry 1 is 8
 Entry 2 is 2
```

```
The original array
 Entry 0 is 6
 Entry 1 is 5
 Entry 2 is 6


The sorted array
 Entry 0 is 5
 Entry 1 is 6
 Entry 2 is 6


The sorted reversed array
 Entry 0 is 6
 Entry 1 is 6
 Entry 2 is 5
```

```
The original array
 Entry 0 is 7
 Entry 1 is 5
 Entry 2 is 0


The sorted array Entry 0 is 0
 Entry 1 is 5
 Entry 2 is 7


The sorted reversed array
 Entry 0 is 7
 Entry 1 is 5
 Entry 2 is 0
```

IV. Rubric and Submission

Item	Points
Class `ArrayProcessor` has three methods with correct input and output types	40
Class `ArrayProgram` correctly creates an array with 3 random integer entries	20
Class `ArrayProgram` correctly invokes the sort, print, and reverse methods in the `ArrayProcessor` class	20
Method sort3IntArray correctly sorts an array	10
Method reverse3IntArray correctly reverses sorted array	10
Code is well formatted and naming conventions are observed	10
Total	100

Create a zip file that contains your two .java files and submit the zip file via Canvas.