CS161 Fall 2015 Programming Exam

Write your name above and return this paper to your lab instructor when you submit your exam. You must finish and submit your work using Check-In by the end of this lab.

Do your own work. Only look at your own computer screen. Do not browse, email, or use cell phones and other digital technology other than the computers in the lab.

During the exam you may use your Java text book (or ebook version if you let your instructor know), the Java API documentation website and the course website, but no other web pages.

Your grade will be based on:

- Correctness of each method

- Clarity of code and comments. Your code should be easy to read, and if there are non-obvious steps, they should be explained with comments.

- Object-oriented design: Does your code capture the idea of encapsulation

- Testing: Does your main method adequately test all aspects of your code?

You will write four java classes for keeping track of storms. The names of the classes are Weather, Storm, Tornado, and Hurricane. The classes Tornado and Hurricane must extend the class Storm, to allow an instance of Weather to use a single ArrayList to hold instances of Tornado and Hurricane.

Your classes must implement the following methods.

Weather: class that is the primary application that you will run.

> **constructor:** Creates an empty ArrayList for holding items of type Storm.
>
> **addStorm:** Add a new Storm, either a Hurricane or a Tornado.
>
> **toString:** Returns a string representing all storms that have been added.
>
> **main:** A main method that tests each method in this class.

Storm: parent class of Tornado and Hurricane.

> **constructor:** Has one int argument for the intensity of a storm. Assigns this value to a protected instance variable.
>
> **getIntensity:** Returns the intensity.
>
> **setIntensity:** Changes the value of intensity. Intensity can be from 0 to 10 (including 10). If the argument to this method is less than 0, change it to 0. If it is greater than 10, set it to 10.
>
> **toString:** Returns a string like `Storm: 3` where 3 is its intensity.
>
> **main:** A main method that tests each method in this class.

Tornado: child class of Storm

> **constructor:** Has two arguments, a String for the name of the state where the Tornado is located, and an int for its intensity. Call the parent class constructor to assign the intensity. Assign the state to a protected instance variable.
>
> **getState:** Returns the state.
>
> **setState:** Changes the value of the state.
>
> **toString:** Returns a string like `Tornado: Oklahoma 3`.
>
> **main:** A main method that tests each method in this class.

Hurricane: child class of Storm

> **constructor:** Has three arguments, a String for the name of the Hurricane, an int for its intensity, and a boolean with a value of true if the Hurricane is over land. Call the parent class constructor to assign the value of intensity. Assign the remaining values to protected instance variables.

> **getName:** Returns the name.

> **setName:** Changes the value of the name.

> **getLand:** Returns true if the hurricane is over land, false otherwise,

> **setLand:** Changes the value of the over-land instance variable.

> **toString:** Returns a string like `Hurricane: Sandy 3 false`.

> **main:** A main method that tests each method in this class.

Remember, you are required to implement a `main` method in every class that tests the methods of that class. After your tests in the main methods of each class work, change the main method in your Weather class to perform the following steps.

1. Add these to your Sky in the order given.

   - A Hurricane named Sandy with an intensity of 5 that is not over land.
   - A Tornado in the state of Missouri with an intensity of 3.
   - A Hurricane named Wow with an intensity of 10 that is over land.
   - A Tornado in the state of Oklahoma with an intensity of 4.
   - A Hurricane named Bob with an intensity of 3 that is not over land.

2. Print the weather.

3. Increase by 1 the intensity of the second storm (index 1) in the ArrayList of storms. Use the get method for ArrayList and get and set methods for Storms.

4. Print the weather again.

5. Print out all storms with an intensity that is greater than 4, using the getIntensity method.

Here are some reminders about how to use ArrayLists.

- Remember to `import java.utils.ArrayList;`

- Constructing an instance: `ArrayList<Type> list = new ArrayList<Type>();`

- Add an object `o` of type Type doing `list.add(o)`

- Get element at index `i` by doing `list.get(i)`

When you are done, combine your files into a **jar file** named `weather.jar`

`jar cvf weather.jar *.java`

and check-in your jar file using the Checkin page at CS161 web site.

And you must turn in this exam form with your name on it to your instructor before you leave recitation.