Name: ______________________________ Section: __________

CSSE 220—Object-Oriented Software Development

Exam 1 – Part 2, Mar. 28, 2014

Allowed Resources on Part 2. Open book, open notes, and computer. Limited network access. You may use the network only to access your own files, the course Moodle and Piazza sites (but obviously don't post on Piazza) and web pages, the textbook's site, Oracle's Java website, and Logan Library's online books.

Instructions. *You must disable Microsoft Lync, IM, email, and other such communication programs before beginning part 2 of the exam. Any communication with anyone other than the instructor or a TA during the exam may result in a failing grade for the course.*

You must actually get these problems working on your computer. Almost all of the credit for the problems will be for code that actually works. There are several different small methods to write, so you can get a lot of partial credit by getting some of them to work. If you get every part working, comments are not required. If you do not get a method to work, comments may help me to understand enough so I can give you (a small amount of) partial credit.

Begin part 2 by checking out the project named *Exam1-201430* from your course SVN repository. (Ask for help immediately if you are unable to do this.)

When you have finished a problem, and more frequently if you wish, **submit your code by committing it to your SVN repository**. We will check commit logs, so you must be careful not to commit anything after the end of the exam. For grading, we will ensure that the included JUnit tests have not been changed.

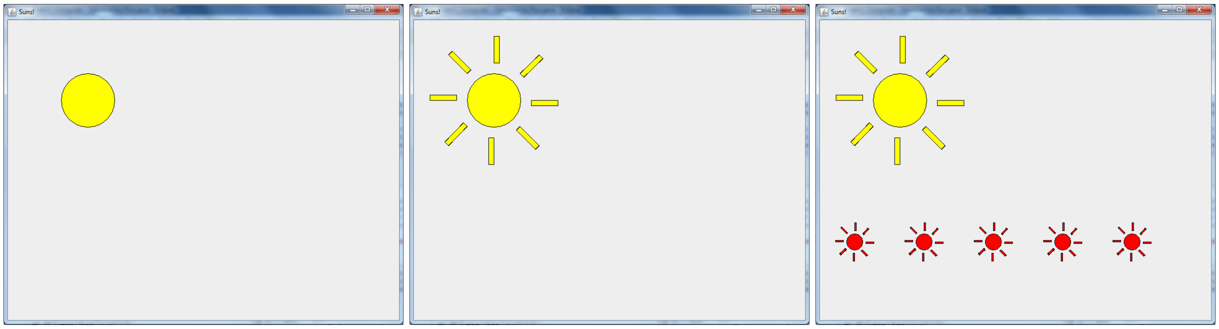
Part 2 is included in this document. **Do not use non–approved websites like search engines (Google) or any website other than those above.** Be sure to turn in the these instructions, with your name written above, to your exam proctor. You should not exit the examination room with these instructions.

Part 2—Computer Part

Problem Descriptions

Part A: 5 Small Problems (30 points) Implement the code for the 5 functions in SmallProblems.java – each problem is worth 6 points. Instructions are included in the comments of each function. Unit tests are included in SmallProblemsTest.java.

Part B on next page



Post Step 4 (left). Post Step 5 (center). Post Step 7 (right).

These images are in your repositories. Note that the color may be washed out in the images, just use Color.RED and Color.YELLOW (or constants where you can) and it will be correct.

Part B: Suns (35 points)

Read over all these instructions carefully. Make sure you understand completely what functionality you have to implement before you start coding. Ask if any part of the instructions are unclear.

Implement the SunComponent and Sun classes. The SunComponent is responsible for creating Sun objects and telling them when to draw. The Sun object is responsible for drawing the Sun's circles and rays to the screen.

A few details about how to draw the Sun:

- The x and y values stored in a Sun make the upper left point of the Sun when drawn.
- The space between the circle of a Sun and it's rays is equal to 20% of the Sun's circle's diameter: (RAY_DISTANCE_FROM_SUN_SCALE)
- The length of a ray is 50% of a Sun's diameter: (RAY_LENGTH_SCALE)
- The width of a ray is 10% of a Sun's diameter: (RAY_WIDTH_SCALE)
- The circleDiameter field is the diameter of the circle in the center of the Sun, not the entire size of the Sun.

Step 1 In the SunComponent class, find TODO: Step 1 and follow the instructions provided.

Step 2 In the SunComponent class, find TODO: Step 2. Use a default constructor to create a new instance of the Sun class.

Step 3 In the Sun class, find TODO: Step 3. Implement the default constructor of the Sun class, using the following default information (also see constants in Sun class):

- X value should be 100.0
- Y value should be 100.0
- Diameter should be 100.0
- Color should be Yellow

You may do other operations in the constructor if you think they will be useful to you later.

Step 4 In the Sun class, find TODO: Step 4. Implement the circle drawing portion of the drawOn method. When done with step, run your program and make sure it appears like the left image from Step 4 above.

Step 5 In the Sun class, find TODO: Step 5. Implement thedrawRays method of the Sun class. Next, call this method from drawOn. When done with step, run your program and make sure it appears like the center image on Page 3.

Step 6 In the Sun class, find TODO: Step 6. Implement a constructor with the specified parameters.

Step 7 In the SunComponent class, find TODO: Step 7. Implement a loop that creates an array of Sun objects and draws them, using the following information (also see constants in SunComponent class):

- Little Suns start at x = 50
- Little Suns have a circle diameter of 30.0
- Little Suns are yellow

When you have completed this step, run your program. It should appear like the Step 7 image on the right on Page 3.

Part A - Small Problems	**Points**	**Earned**
isSecondToLastCharacterV	6	______
combineArrays	6	______
insertAtMiddle	6	______
removeStringsGreaterThanLength4	6	______
removeItemsFromList	6	______
Part B - Sun and SunComponent		
Step 1 functionality	2	______
Step 2 functionality	5	______
Step 3 functionality	5	______
Step 4 functionality	5	______
Step 5 functionality	5	______
Step 6 functionality	8	______
Step 7 functionality	5	______
Computer Part Subtotal	**65**	______