CMSC 132 Quiz 2 Worksheet

The second quiz for the course will be on Wed, Sep 16. The following list provides more information about the quiz:

- The quiz will be a written quiz (no computer).
- The quiz will be in your lab (discussion) session.
- Closed book, closed notes quiz.
- Answers must be neat and legible. We recommend that you use pencil and eraser.

The following exercises cover the material to be included in this quiz. Solutions to these exercises will not be provided, but you are welcome to discuss your solutions with the TA or instructor during office hours.

Exercises

1. What is the difference between a checked and an unchecked exception?

2. Which of the following are valid, assuming the Animal class is an abstract class and Elephant is a concrete class (i.e., it implements any abstract methods associated with the Animal class). You can assume the Elephant class has a default constructor.
   a. Animal a = null; // VALID/INVALID
   b. Animal a = new Animal(); // VALID/INVALID
   c. Animal a = new Elephant; // VALID/INVALID
   d. Elephant b = new Elephant(); // VALID/INVALID
   e. Elephant c = new Animal(); // VALID/INVALID

3. When would you use an abstract class rather than an interface?

4. When would you use an interface rather than an abstract class?

5. Define an abstract class called Oven that has an abstract method called turnOnOff. The class also has an instance variable representing a serial number. Implement the following subclasses:
   a. MicrowaveOven – Defines a method called “setTimer” that sets the time the microwave will be active.
   b. ConventionalOven – Defines a method called “setTemperature” that sets the temperature for the oven.

6. The Clock class is defined as follows:
   public class Clock {
       private int hours, minutes, seconds;
   }
   a. Implement an equals method that relies on instanceof.
   b. Implement an equals method that relies on getClass().
   c. Implement a clone method for the class.

7. What is test coverage?

8. What is the difference between a static and a non-static initialization block?