CSE2211: OOP Java

Quiz #7 (Inheritance and Interfaces, 10.12.2018)

1. Suppose the class Employee is declared as follows:

```
public class Employee {
   private String name;
   private double baseSalary;
   public void setName(String newName) { . . . }
   public void setBaseSalary(double newSalary) { . . . }
   public String getName() { . . . }
   public double getSalary() { . . . }
}
```

a) Declare a class **`Manager`** that inherits from the class **`Employee`** and adds an instance variable **`bonus`** for storing a salary bonus. Omit constructors and methods.

b) Which instance variables does the **`Manager`** class have?

c) In the **`Manager`** class, provide the method header (but not the implementation) for a method that overrides the **`getSalary`** method from the class **`Employee`**.

d) Which methods does the **`Manager`** class inherit?

2. What is wrong with the following implementation of the display method?

```
public class ChoiceQuestion {
   . . .
   public void display() {
         this.display();
         for (int i = 0; i < choices.size(); i++) {
               int choiceNumber = i + 1;
               System.out.println(choiceNumber + ": " +
                                                choices.get(i));
         }
   }
}
```

3. Make a class **Employee** with a **name** and **salary**. Make a class **Manager** inherit from **Employee**. Add an instance variable, named **department**, of type **String**. Supply a method **toString** that prints the manager's name, department, and salary. Make a class **Executive** inherit from **Manager**. Supply appropriate **toString** methods for all classes.

4. The **java.awt.Rectangle** class of the standard Java library does not supply a method to compute the area or perimeter of a rectangle. Provide a subclass **BetterRectangle** of the **Rectangle** class that has **getPerimeter** and **getArea** methods. *Do not add any instance variables*. In the constructor, call the **setLocation** and **setSize** methods of the **Rectangle** class.

5. Suppose you want to use the average method to find the average salary of an array of **Employee** objects. What condition must the **Employee** class fulfill?

6. If both **BankAccount** and **Country** implement the **Measurable** interface, can a **Country** reference be converted to a **BankAccount** reference?

7. Why would you use an inner class instead of a regular class?

8. When would you place an inner class inside a class but outside any methods?

9. Consider this top-level and inner class. Which variables can the **f()** method access?

```
public class T {
     private int t;
     public void m(final int x, int y) {
          int a;
          final int b;
          class C implements I {
               public void f() {
                    .........
               }
          }
          final int c;
          .........
     }
}
```