CSSE 221: Fundamentals of Software Development Honors

Key

Programming assignment (due by Saturday, 11:59pm)

Finish TBA, as stated in the specification, being sure you have checked in your final, fully-documented copy to the repository.

Written homework to do before week 5 (due before beginning of class on Monday, Oct 1).

Answers are to be written in your own words! As usual, please bring hardcopy for written answers to class to hand in on Monday.

1. Complete all the reading.

The next 4 questions are covered in the reading from the last couple weeks. They'll help prepare you for Exam 1 next week.

2. (5 pts) Weiss 1.15 (Weiss uses this construct in later chapters, so you should recognize it)

   SOLUTION:

   ```
   while (true)

       statement
   ```

3. (5 pts) Weiss 2.6 (toString). **Explain** the output. (You can run this if you want.)

   SOLUTION:

   System.out.println(x + ‘ ‘ + y); gives the output 44.

   System.out.println(x + “ “ + y); gives the output of “5 7”

   This is because Java reads characters as integers and the integer value of ‘ ‘ is 32. The second gives 5 7 because they are being concatenated with/to a string rather than added to an integer value.

4. (10 pts) Weiss 2.11 (resize array in method) Again, you can run the code and verify that it doesn't work, but you need to explain why not.

   SOLUTION:

   Because things in Java are passed by value, the array's location is passed into the method. When you attempt to reassign it to a new location, it works within the method, but once you exit the method the original item has not changed.

5. (20 pts) Weiss 4.6 (inheritance) – Make assumption all in separate packages

   SOLUTION:

   NOTE: This question was thrown out as it had too many interpretations. Full credit was given for any attempt on a problem. Below are the answers given you put each in a separate package.

   a. Which accesses are illegal? - bProtected, bPrivate, dPrivate
   b. Make main a method in Base. Which accesses are illegal? – dPrivate
   c. Make main a method in Derived. Which accesses are illegal? –bProtect, bPrivate
   d. How do they change if protected is removed from line 4?
      1. Main in Tester – Same
      2. Main in Base – Same
      3. Main in Derived – Same
   e. Write a three parameter constructor for Base. Then write a five parameter constructor for Derived.
      1. public Base(int bPublic, int bProtect, int bPrivate) {
         this.bPublic = bPublic;
         this.bProtect = bProtect;
         this.bPrivate = bPrivate;
         }
      2. public Derived(int bPublic, int bProtect, int bPrivate, int dPublic, int dPrivate) {
         super(bPublic, bProtect, bPrivate);
         this.dPublic = dPublic;
         this.dPrivate = dPrivate;
         }
   f. The class Derived consists of 5 integers. Which are accessible to the class Derived? – all except bPrivate
   g. A method in the class Derived is passed a Base object. Which of the Base object members can the derived class access? - All except bProtected, bPrivate

PLEASE LEAVE TO BE GRADED BY 2ND SECTION TAs (long story, but trust us on this, you don't want to deal with this)

6. (10 pts) Weiss 4.9 (interface vs. abstract classes)

   SOLUTION:

   An interface is a collection of method headers that specify what an inheriting class must implement. It can also contain public static final fields for the class. Private members cannot exist in an interface because inheriting classes cannot see them so they can never be used. An interface differs from an abstract class in that it is implemented rather than extended and it cannot contain any implementation.

7. (10 pts) Weiss 4.14 (local and anonymous)

   SOLUTION:

   A local class is a class that is created within a method and has no visibility modifier. An anonymous class has no name and is mostly useful for implementing short function objects.

8. (20 pts) Weiss 4.19 (polymorphism)

   SOLUTION

   a. First Three Lines Correct, Last Incorrect – a Base[] cannot be cast as a Derived[]
   b. First three lines are correct, last line not necessary
   c. No unnecessary.
   d. First line OK, Second line is wrong because a Derived[] array cannot contain members of type Base

9. (40 pts) Weiss 4.29 and 4.30 (function objects). On 4.30, EqualsK should implement your interface from the first problem. Add extra tests to your main method to create some different equalsK objects and pass them, along with an interesting array, to the countMatches() method. Hints: you may find it helpful to draw an analogy with the SimpleRectangle example I presented in class: countMatches (like findMax) is the method that takes an array and a function object as parameters. EqualsZero (like findMaxByWidth) is specific function object. ??? (like Comparator) is the function object interface: you pick the name of the interface you'll write. Note: please call your project: **WeissCountMatches** and do a Team>Share to submit it to the repository for grading.