Problem Set 1

Assigned: Sept. 7
Due: Sept. 19

In homework assignments that ask you to write Java code, such as exercises 1, 2, and 3 below, you only have to write the code on your submitted homework; you don’t have to get it working. Therefore, you will only lose a small part of the credit for minor errors that would actually prevent it from running, like trivial syntax errors (e.g. a missing semicolon). Of course, a very good way to be sure you’ve got the right answer is to get it working.

On problems that ask you what a given piece of code will do, such as exercise 4, it is 100% OK to download it, run it, and see what it does. However it is critical that you should understand why it does what it does; otherwise, you will not be able to answer the question on the exam, where you don’t have the option of running it.

Exercise 1

A. Add to the class Rectangle a method void scaleDest(double x) such that q.scaleDest(x) destructively expands both sides of Rectangle q by a factor of s. That is, the object q is changed. For instance if q is a 3 × 5 rectangle, then calling q.scaleDest(2.0) changes q into a 6 × 10 rectangle.

B. Add to the class Rectangle a method Rectangle scaleNonDest(double x) such that q.scaleNonDest(x) nondestructively creates and returns a new rectangle whose sides are s times the sides of q. The object q itself should be unchanged. For instance if q is a 3 × 5 rectangle, then calling q.scaleNonDest(2.0) creates a new 6 × 10 rectangle.

C. Write a small driver, comparable to TestRectangle, that illustrates the difference between the two methods.

Exercise 2

A. Write a class Square that extends Rectangle with the following features:
   i. The constructor Square(double side) initializes both the xSpan and the ySpan fields to be equal to side.
   ii. There are methods getSide() and setSide(double side) that do the obvious things.
   iii. If q is a Square then calling q.setSpans(x,y) will print out a warning message, "You may not use setSpans to set the sides of a square!" and have no effect.

B. Write a class LocatedSquare that extends Square in the same ways that LocatedRectangle extends Rectangle.

C. Can you do part (B) by having LocatedSquare extend both Square and LocatedRectangle?
Exercise 3

Add to the class Person the following fields and methods.

A. A data field bff, of class Person.

B. A data field bffCount of type int. The field bffCount keeps track of the number of different BFF’s the Person has gone through since the object was created. It is initialized to 0.

C. Getters getBff() and getBffCount().

D. A setter setBff(Person q). If p and q are Persons then calling p.setBff(q) should

i. If q is null, print an error and exit. (That is, exit the method; do not terminate the entire program.)

ii. Check that both p and q currently have no BFF. If either of them does, then it should print an error message and have no effect.

iii. Set p’s BFF to be q and q’s BFF to be p.

iv. Increment bffCount for both p and q.

E. A method unBFF(). If you call p.unBFF() and q is currently p’s BFF, then both p’s BFF and q’s BFF should be set to null. If p has no BFF, then nothing should happen; the program should not crash.

Exercise 4

Consider the code for Hwk1Ex4.java, linked on the course web site.

A. What does this output?

B. Explain the output.