1. Suppose that rectangles in a Cartesian plane are represented by a class Rectangle that has 4 private fields as shown:

class Rectangle

{

private int left; //x-coordinate of left edge

private int bottom; //y-coordinate of bottom edge

private int width; //width of rectangle

private int height; //height of rectangle

}

1. Write a constructor method that has four parameters representing the 4 fields of the class. The constructors should replace any negative length parameters with zero.
2. Create the accessor and mutator to get and set the instance variables.
3. Write a toString method for the Rectangle class. For the rectangle with lower left corner located at (3, 2) and having width 4 and height 5, the method should return “base: (3, 2) w: 4 h: 5”.
4. Write a class method, intersection, that has two Rectangle parameters. The method should return the rectangle formed by the area common to the two rectangles. If they do not intersect, the method should return the rectangle whose instance fields are all zero. If the rectangles touch, the method should return a “rectangle” of zero width or zero length.
5. Write a class method, totalPerimeter, that has two Rectangle parameters. The method should return the total perimeter of the figure formed by the two rectangles. It should only count as part of the perimeter those portions of the edges of a rectangle that are on the exterior of the resulting figure. If one rectangle is completely contained by the other, then the method should return the perimeter of the outer rectangle. If the rectangles do not intersect, the method should return the sum of the individual perimeters.
6. Write an instance method, contains, that has one explicit parameter of type Rectangle. The method should return true if every point of the rectangle determined by the explicit parameter is on or within the rectangle determined by the implicit parameter. It should return false otherwise.
7. Create a main program to test the Rectangle class in question #1 and continue to ask user for another set of rectangle until
   1. Ask user for the left, bottom, width and height of rectangles.
   2. Write a class method, area, that returns the area of a rectangle.
   3. Write a class method, perimeter, that returns the perimeter of a rectangle.

Example output:

Area of A: 123.45

Perimeter of A: 34

Area of B: 234.56

Perimeter of B: 42

Intersection of A&B: base: (3,2) w:4 h:5

Total Perimeter A&B: 64

A contains B: False