CS 180 Problem Solving and OO Programming

Fall 2011

Recitation Week 11
Interfaces

You are given the following interface.

```
public interface SimpleGeometry{

//  Return length of the straight line joining  Point objects p  and q.
public double length(Point p, Point q);


  // Return true if the three Point objects form a right angled triangle.
 // Example usage: boolean rt=rightAngledTriangle(p, q, r, 0.1E-6);
 // If p-q is the hypotenuse, d1, d2, and d3 denote, respectively, the lengths of
// lines joining points p-q, a-r, and p-r, then rt will be true only when
// d1-sqrt(d2*d2+d3*d3)<0.1E-6.
public boolean rightAngledTriangle(Point p, Point q, Point r, double tolerance);

  // This returns true when all sides of the triangle are equal subject to
   // the error given by tolerance.
  public boolean isoscelesTriangle(Point p, Point q, Point r, double tolerance);
}// End of SimpleGeometry
```

(a) Create a class named **Point**. This class will be used to create **Point** objects. Each **Point** object contains the x and y coordinates of a point in 2-dimensions.
(b) Create a class named **Triangle**. This class must implement the `SimpleGeometry` interface.
(c) Add `main()` to the **Triangle** class. In `main()` create three **Point** objects named a, b, and c, each denoting, respectively, the points (0,0), (1,0), and (1,1).
(d) Use the `length()` method to find and print the length of the line joining points a and b.
(e) Use the `rightAngledTriangle()` method to determine whether or not the triangle with vertices a, b, and c is a right angle triangle. Assume a tolerance of 0.1E-6.
(f) Use the `isoscelesTriangle()` method to determine whether or not the triangle with vertices a, b, and c is a isosceles triangle. Assume a tolerance of 0.1E-6.

<End of Problems for Week 11>