

Lab #5: Java Introduction

This lab is intended to give you your first exposure to Java programming. You will implement a few basic classes relating to 2D shapes and put them together.

1. Point Class

Create a Point class (see the Point2D class in the lecture notes) with an x and y coordinate, both doubles. It should have the following methods:

- a) double distanceBetween(Point p) – returns the straight line distance between this Point, and the Point p.
Hint: distance is given as $d = \text{sqrt}((x_2 - x_1)^2 + (y_2 - y_1)^2)$
- b) double getX() – returns this point's x component as a double (accessor method)
- c) double getY() – returns this point's y component as a double (accessor method)
- d) void move(double dx, double dy) – move this point by the vector (dx,dy)
- e) void move(Point p) – move this point by the vector provided by p (i.e., this.x + p.x, this.y + p.y)
- f) default constructor: public Point() – creates a point with an x, y coordinate of 0,0.
- g) Constructor: public Point(double initx, double inity) – creates a point using the provided initx and inity for the x and y values.
- h) void print() – prints the values for this Point out in a nicely formatted way

2. Circle Class

Create a Circle class with a centre that is a Point2D object. It should also have a double radius field. It should have the following methods:

- a) boolean pointInside(Point p) – returns true if the Point p is within the radius of the circle.
- b) boolean overlap(Circle C) – return true if the two circles overlap at all
- c) double perimeter() – returns the perimeter of the circle
- d) double area() – returns the area of the circle
- e) default constructor: public Circle() – creates a circle with a centre point of 0,0 and a radius of 1
- f) constructor: public Circle(Point p, double r) – creates a circle using p as the centre, and r as the radius
- g) constructor: public Circle(double x, double y, double r) – creates a circle using x, y as the centre Point and r as the radius
- h) constructor: public Circle(Point p1, Point p2) – creates a circle using p1 as the centre point, and p2 as a point on the perimeter of the circle to determine the radius.
- i) Accessor methods: Point getCentre(), double getRadius()
- j) void print() – prints the values for this Circle out in a nicely formatted way

3. Quad class

Create a Quad class (for rectangles, squares and other quadrilaterals) that has an array of 4 Points for its four corners. It should have the following methods:

- a) boolean pointInside(Point p) – returns true if the Point p is inside the quad.
- b) double perimeter() – returns the perimeter of the quad
- c) double area() – returns the area of the quad
- d) default constructor: public Quad() – creates a Quad with the four corners at (0,0), (0,1), (1,1) and (1,0)
- e) constructor: public Quad(Point p1, Point p2, Point p3, Point p4) – creates a Quad using the 4 Points provided as the corners
- f) void print() – prints the values for this Quad out in a nicely formatted way

4. Shapes class

The final class will be the “main” class for this program - this will be your class that has the main method. It should demonstrate usage of the bulk of the methods above. Specifically, it should create some points, use them to create Circles and Quads to demonstrate the various constructors. It should then use the movePoint methods of the Points (use your Point.print() method to demonstrate that they moved), and demonstrate the perimeter and area methods of the Circle and Quad.

Each class should be in its own .java file, and the name of this should match the class name.

Submit via email to Mike upon completion. Due date is Thursday, April 16, 2009. Late assignments lose 10% per day late, receive zero after 5 days late (April 23).