CSE 3302/5307 Lab Assignment 4 - corrected

Due December 4, 2014

Goals:

Understanding of JavaScript prototypal inheritance.

Requirements:

1. Implement a class hierarchy (using `Object.create()`) representing

   a. (1) Simple, convex (non-crossing) polygons as a list of counter-clockwise points (last point wraps around to first point).
   b. (2) Rectilinear rectangles as four values: lowX, highX, lowY, highY.
   c. (3) Rectilinear squares as the coordinates of the leftmost lowest point and the length of a side.
   d. (4) A triangle with three counter-clockwise points.

   and including methods to construct an instance, compute the area, and to indicate whether a provided point is inside (including on the border) of the region.

2. The input is to be taken from an html textarea (in response to a button click) and the output written to an html textarea:

   a. A line indicating the number of objects that follow, one object per line. References to the objects should be placed in a 0-based array.

      1. The number 1, the number of counter-clockwise points, and then pairs of x/y coordinates.
      2. The number 2, followed by the four bounding values.
      3. The number 3, followed by an x/y pair for the leftmost lowest point and then the length of both sides.
      4. The number 4, followed by three x/y pairs for the three counter-clockwise points.

   b. A line indicating the number of command lines that follow. The command lines may be formatted as:

      1. The number 1 and an object's subscript to indicate that the area should be computed.
      2. The number 2, an object's subscript, and an x/y pair to be tested for being inside.
      3. The number 3 and one of the four values above for retrieving the number of instances in that class. (a square is a rectangle, while rectangles and triangles are simple polygons)
      4. The number 4 and an object's subscript for retrieving the bounding box, a rectilinear rectangle.

3. For up to 50 extra credit lab points (not included in the final grade "curve"), include the following:

   a. Number 5, rectilinear ellipse as four values: lowX, highX, lowY, highY.
   b. Number 6, circle as the coordinates of the center along with the radius.
   c. Number 7, arbitrary polygon as a list of points. An area method is not expected.
   d. A circle is an ellipse and a region is an abstract class including all mentioned classes. Arbitrary polygons include simple polygons.
   e. The region abstract class should have a default area computation based on randomized ("Monte Carlo") use of the bounding box and point-inside methods.
   f. Command 3 may have 8 as its argument to indicate that the total number of shapes is needed.

4. Submit your zipped `.html/.js` files on Blackboard by 1:45 p.m. on December 4.

Getting Started:

1. All values will be provided as strings corresponding to integers. Conventional cartesian coordinates, not window coordinates.

2. If a polygon is convex, then any three consecutive points will make a left turn.

3. If invalid input is detected, give an error message and stop.

4. I will provide sample html.

5. `http://ranger.uta.edu/~weems/NOTES5311/notes17.pdf` has basic information regarding geometry.