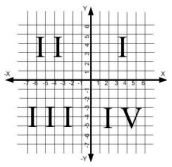
CSAP Point Class Practice FRQ

The coordinates of a point are a pair of numbers that define its exact location on a two-dimensional plane and are written as an ordered pair. The X and Y axes, illustrated on the right, divides the plane into four quadrants and are sometimes labeled with Roman numerals: I, II, III and IV.



The distance *d* between the points (x_1, y_1) and (x_2, y_2) is given by $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$

The midpoint *m* between the points (x_1, y_1) and (x_2, y_2) is given by $m = \left(\frac{x_1 + x_2}{2}, \frac{y_1 + y_2}{2} \right)$

Assume the following appears in a class other than APPoint. The code segment shows an example of using the APPoint class.

```
APPoint A = new APPoint(-3.0,2.0);
APPoint B = new APPoint(1.0,-1.0);
APPoint origin = new APPoint(0.0,0.0);
System.out.println (origin.getQuadrant());          //O
System.out.println (A.getQuadrant());               //II
System.out.println (A.getDistance(B));              //5.0
System.out.println (A.getMidX(B));                  //-1.0
System.out.println (A.getMidY(B));                  //0.5
System.out.println (A.getY());                      //2.0
System.out.println (A);                             //(-3.0,2.0)
```

Write the `APPoint` class. Your implementation must include a constructor that has two `double` parameters representing the x-coordinate and the y-coordinate. Your class must include the following methods:

- `getX()` — returns the x-coordinate
- `getY()` – returns the y-coordinate
- `getQuadrant()` – returns the quadrant expressed as a Roman numeral . If the point resides on an axis, `getQuadrant()` returns the letter O
- `getDistance(APPoint p)` – returns the distance of the line between `(x,y)` and the coordinates of point `p`. The distance is expressed as a `double`
- `getMidX(APPoint p)` – returns the x-coordinate of the midpoint of the line between the `x` and `y` of the executing object and Point `p`, and is expressed as a `double`
- `getMidY(APPoint p)` – returns the y-coordinate of the midpoint of the line between the `x` and `y` coordinate of the executing object and point `p` and is expressed as a `double`
- `toString()` – returns the `x` and `y` coordinates written as an ordered pair