```java

public class Circle {
    /** The radius of the Circle */
    private int radius;
    /** The x coordinate of the center of the Circle */
    private int x;
    /** The y coordinate of the center of the Circle */
    private int y;

    /**
     * Constructs a Circle with a radius of 1 centered at (0, 0)
     */
    public Circle(){
        radius = 1;
        x = 0;
        y = 0;
    }

    /**
     * Constructs a Circle with a radius of 1 centered at (x, y)
     * If x of y is not between -100 and 100 it will be set to 0
     * @param x The x coordinate of the center of the Circle
     * @param y The y coordinate of the center of the Circle
     */
    public Circle(int setX, int setY){
        radius = 1;
        x = setX;
        y = setY;
        if(y < -100 || y > 100) y = 0;
        if(x < -100 || x > 100) x = 0;
    }

    /**
     * Constructs a Circle with a radius of radius centered at (x,
  y)
     * If x of y is not between -100 and 100 it will be set to 0
     * If radius is 0 or negative, the radius of the circle will
  be 1
     * @param x The x coordinate of the center of the Circle
     * @param y The y coordinate of the center of the Circle
```

```
39      */
40     public Circle(int setX, int setY, int setRadius){
41         radius = setRadius;
42         if(radius < 0) radius = 1;
43         x = setX;
44         y = setY;
45         if(y < -100 || y > 100) y = 0;
46         if(x < -100 || x > 100) x = 0;
47     }
48
49     /**
50      * Gets the x coordinate of the center of the Circle
51      * @return the x coordinate of the center of the Circle
52      */
53     public int getX(){
54         return x;
55     }
56
57     /**
58      * Gets the y coordinate of the center of the Circle
59      * @return the y coordinate of the center of the Circle
60      */
61     public int getY(){
62         return y;
63     }
64
65     /**
66      * Gets the radius of the Circle
67      * @return the radius of the Circle
68      */
69     public int getRadius(){
70         return radius;
71     }
72
73     /**
74      * Gets the area of the Circle
75      * @return the area of the Circle
76      */
77     public double getArea(){
78         return Math.PI*radius*radius;
```

```
79      }
80
81      /**
82       * Gets the circumference of the Circle
83       * @return the circumference of the Circle
84       */
85      public double getCircumference(){
86          return 2*Math.PI*radius;
87      }
88
89      /**
90       * Gets the quadrant the Circle's center is in
91       * @return the quadrants (1, 2, 3 or 4) of the Circle's
    center, 0 if the center on either axis
92       */
93      public int getQuadrant(){
94          if(y > 0 && x > 0) return 1;
95          else if(y > 0 && x < 0) return 2;
96          else if (y < 0 && x < 0 )return 3;
97          else if (y < 0 && x > 0) return 4;
98          return 0;
99      }
100
101     /**
102      * Changes the x coordinate of the center of the circle if
    newX is between -100 and 100 inclusive
103      * @param newX the new x coordinate of the circle
104      * @return true if the x coordinate was changed
105      */
106     public boolean setX(int newX){
107         if(newX < -100 || newX > 100) return false;
108         x = newX;
109         return true;
110     }
111
112     /**
113      * Changes the y coordinate of the center of the circle if
    newY is between -100 and 100 inclusive
114      * @param newY the new y coordinate of the circle
115      * @return true if the y coordinate was changed
```

```
116      */
117     public boolean setY(int newY){
118         if(newY < -100 || newY > 100) return false;
119         y = newY;
120         return true;
121     }
122 }
123
```