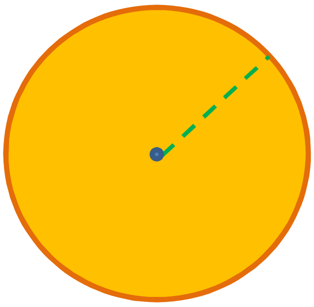
Practice Exercise #13: Ball with centre

http://www.comp.nus.edu.sg/~cs1020/4_misc/practice.html



Reference: Week 3 OOP Part 2

Objectives:

1. Object-Oriented Programming
2. Using the API **Point** class
3. Writing user-defined class, employing the following features:
   a. Overloaded constructors
   b. Using "**this**" in constructor
   c. Overriding methods **toString()** and **equals()**

Task statement:

Write **CentredBall.java** to define a **CentredBall** class which is an enhancement of the **MyBall** class discussed in lecture, by including a new instance attribute, the centre of the ball:

- private Point centre;

This requires the use of the **Point** class.
(Look up on http://docs.oracle.com/javase/7/docs/api/java/awt/Point.html)

The class should provide 3 constructors:

- Default constructor **CentredBall()** that creates a yellow ball of radius 10.0, centred at the origin (0, 0).
- **CentredBall(String colour, double radius, Point centre)**
- **CentredBall(String colour, double radius, int xCoord, int yCoord)** where xCoord and Ycoord are the coordinates of the centre of the ball

You should add appropriate accessor and mutator methods, and the overriding **toString()** and **equals()** methods in **CentredBall.java**.

Write also a client program **TestCentredBall.java** so that it runs according to the given sample run. The program should perform the following:

- Create 3 **CentredBall** objects
- Compare the first ball with the second
- Print out the largest ball (if there are more than one largest ball, print out the first one among them)

The last line of output as shown in the sample run is to be returned by the **toString()** method.

Sample run:

```
1st ball
Enter colour, radius and centre: red 5.5 82 12
2nd ball
Enter colour, radius and centre: blue 3.1 -10 5
3rd ball
Enter colour, radius and centre: green 20.3 16 32

1st and 2nd balls are not the same.
The largest ball created is:
[colour=green, radius=20.3, centre=(16,32)]
```