

**Homework 3**  
**Due February 4th 11:00 AM**  
**100 points**  
CS 2235  
Data Structures and Algorithms

**1. Create a Point class:**

- a) Create a new class called Point.
- b) It should have 2 protected instance variables X and Y, with integer data type.
- c) Create a default constructor which instantiates Point with X and Y equal to 0.
- d) Create a 2-parameter constructor, which can pass a user defined X and Y, to create a Point object.
- e) Create public getter and setter methods for the 2 protected instance variables.
- f) Demonstrate that your constructors, getter and setter methods, all work by:
  - 1) Creating 2 Point objects, 1 using the default constructor and the other by passing the parameters (Choose your own values for the parameters).
  - 2) Using the getter methods to print the X and Y values for both Points.
  - 3) Change the values of both instance variables of your second Point to 2, using the setter methods and print the new values to the screen.

**2. Create a Shape class:**

- a. Create a new class called Shape. Do this in a new file.
- b. It should have a protected String instance variables called name.
- c. Create a single parameter constructor which can pass a name to create a Shape object.
- d. Create a getter and setter method for this instance variable.

**3. Create a Square subclass:**

- a. Create a subclass for the Shape class called square.
- b. It should have 2 protected Point instance variables.
- c. Create a 3-parameter constructor, which can pass a name, and 2 Point objects to create a Square object.

- d. Create a getter method called `getLength`. This method should take the X values of the two Point objects in the Square, calculate the length of the square and return the value for length. *Hint:* To ensure your height value is always positive, you can use the absolute value method (`Math.abs()`) from the Math library built in to Java(`java.lang.Math`);).
- e. Create a getter method called `getHeight`. This method should take the Y values of the two Point objects in the Square, calculate the height of the square and return the value for height.
- f. Create a getter method called `getArea`. This method should calculate the area using the `getHeight` and `getLength` methods, then return the value of the area. *Hint:* Simply use the syntax `getHeight()` and `getLength()` to obtain the values needed to calculate the area.
- g. Create a getter method called `getPerimeter`. This method should calculate the perimeter using the `getHeight` and `getLength` methods, then return the value of the perimeter.
- h. Demonstrate your class and methods work by creating a Square object using the 2 Point objects you created in Part 1 and a name of your choice. Print the length, height, area and perimeter of your Square object.

### 3. Create a Circle subclass:

- a. Create a subclass for the Shape class called Circle.
- b. It should have 2 protected instance variables, a Point (the center of your circle) and a radius.
- c. Create a 3-parameter constructor which can pass a name, a Point object and a radius to create a Circle object.
- d. Create a getter method which returns the value of the radius called `getRadius`.
- e. Create a getter method that returns the center of your circle. *Hint:* You will be returning a Point object for this method.
- f. Create a getter method called `getArea`. This method should calculate the area using the `getRadius` method and the value 3.14 for Pi. Then return the value of the area. *Note:* Since we are working with Pi, we will need to define our radius as a double. Simply redefine your radius in your method as a double.
- g. Create a getter method called `getCircumference`. This method should calculate the circumference of your circle using the `getRadius` method and the value 3.14. Then, return the value of the circumference.

- h. Demonstrate your class and methods work by creating a Circle object using the first Point object you created in Part 1, a radius of 1 and a name of your choice. Print the name, length, height, area and perimeter of your circle.

Note: Use a separate Java file for the each of the classes, subclasses and testing. In total, you should submit 5 Java files.

### Scoring

1. 10% - Code compiles without errors.
2. 15% - Point class coded with all required instance variables, constructors and methods.
3. 15% - Point class methods and constructors demonstrated correctly.
4. 10% - Shape class coded with all required instance variables, method and constructor.
5. 10% - Square subclass coded with all required instance variables, methods and constructor.
6. 10% - Circle subclass coded with all required instance variables, methods and constructor.
7. 20% - Both, Square and Circle classes demonstrated correctly.
8. 10% - Meaningful comments and header.

### Sample Output:

```
My first point is: 0,0
My second point is: 1,1
My second point is now: 2,2
The name of my square is: My Square
The length of my square is: 2
The height of my square is: 2
The area of my square is: 4
The perimeter of my square is: 8
The name of my circle is: My Circle
The radius of my circle is:1
The center of my circle is at: 0,0
The area of my circle is: 3.14
The circumference of my circle is 6.28
```