Lab 5 – CPS 109

This lab helps you get familiar with classes and inheritance.

1. Write a class called Shape.

   ```
   class Shape:
   ```

2. The Shape class is going to represent a general shape in the 2D plane. A shape is described by a set of points that determine its boundary. Write a class called Point. The point class has two attributes x and y which are unique to each point. So we define them as instance variables, i.e., using the self object. We initialize them in the __init__ mathod.

   ```
   class Point(x, y):
        def __init__(self, x, y):
              self.x = x
              self.y = y
   ```

3. Write the __str__ method for the Point class. The method should return a string in the following format: "(x, y)"

4. Test the point class using the following:

   ```
   p = Point(31, 7)
   print(p)
   ```

5. Add the __init__ method to the shape class. In the method you should initialize the list of boundary points to empty:

   ```
   self.boundary = []
   ```

6. Add a method called set_boundary to the Shape class. The set boundary method accepts a list of points called boundary. In the method you should set self.boundary to the input boundary:

   ```
   def set_boundary(boundary):
   ```

7. Write the __str__ method for Shape. The method should return a string in the following format: "Boundary: (x1, y1), (x2, y2), ..." where the set of points are the ones in self.boundary. Hint1: remember that the point class gives you a string of the form "(x, y)", so you can use that here. For example if self.boundary[0] is the point p = (5, 12) then str(p) is the string "(5, 12)". Hint2: you should loop through all the points in self.boundary to get the final result.

8. Define a class called Rectangle. The Rectangle class will be a subclass of Shape:

   ```
   class Rectangle(Shape):
        pass
   ```

9. Test the rectangle class using the following. The first point is assumed to be the bottom left corner of the rectangle.

```
rect = Rectangle()
rect.set_boundary([Point(0, 0), Point(0, 3), Point(5, 3), Point(5, 0)])
print(rect)
```

10. Define a class called Square. The Square class will be a subclass of Rectangle:

```
class Square(Rectangle):
      pass
```

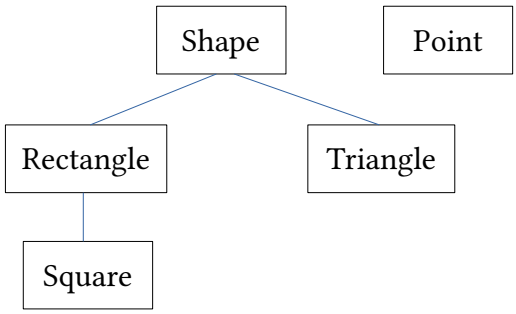
Test the Square class using the following. The first point is assumed to be the bottom left corner of the square.

```
square = Square()
square.set_boundary([Point(0, 0), Point(0, 3), Point(3, 3), Point(3, 0)])
print(square)
```

11. Define a class called Triangle. The Triangle class will be a subclass of Shape. Test the Triangle class using the following:

```
trgle = Triangle()
trgle.set_boundary([Point(0, 7), Point(3, 3), Point(2, 5)])
print(tgle)
```

12. The diagram of the classes now looks like the following:



13. Add the following method to the Square class:

```
def set_side_length(length):
```

The argument length is an integer, based on which the new boundary point of the square will be calculated. Since the first point, i.e., p = self.boundary[0], is the bottom left corner, the point above p, i.e., self.boundary[1], will be (p.x, p.y + length). Figure out the position of the rest of the points. Test your code using the following:

```
square.set_side_length(5)
print(square)
```