

In-Lab Demo - Sep. 18

Programming Pattern

Reference-Typed Arrays
Integer Counter
Breakpoints & Debugger

Programming Pattern

```
public class PointCollector {  
    private Point[] points;  
    private int nop; /* number of points */
```

```
    public PointCollector() {  
        this.points = new Point[100];  
    }
```

```
    public void addPoint(double x, double y) {  
        this.points[this.nop] = new Point(x, y);  
        this.nop++;  
    }
```

```
    public int getNumberOfPoints() {  
        return this.nop;  
    }
```

```
    public Point[] getPointsInQuadrantI() {  
        Point[] ps = new Point[this.nop];  
        int count = 0;
```

```
        /* number of points in Quadrant I */
```

```
        for(int i = 0; i < this.nop; i++) {  
            Point p = this.points[i];  
            if(p.getX() > 0 && p.getY() > 0) {  
                ps[count] = p; count++;  
            }  
        }
```

```
        Point[] q1Points = new Point[count];
```

```
        /* ps contains null if count < nop */
```

```
        for(int i = 0; i < count; i++) {  
            q1Points[i] = ps[i];  
        }
```

```
        return q1Points;
```

↓ knowing the # of q1 points, initialize "q1points" accordingly

```
@Test
```

```
public void test() {  
    PointCollector pc = new PointCollector();  
    assertEquals(0, pc.getNumberOfPoints());
```

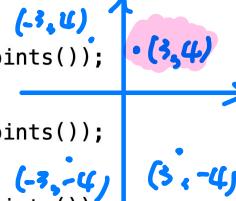
```
    pc.addPoint(3, 4);  
    assertEquals(1, pc.getNumberOfPoints());
```

```
    pc.addPoint(-3, 4);  
    assertEquals(2, pc.getNumberOfPoints());
```

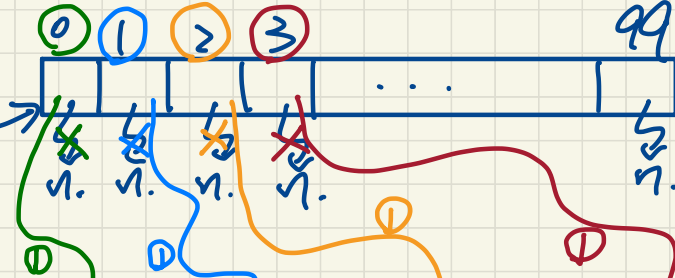
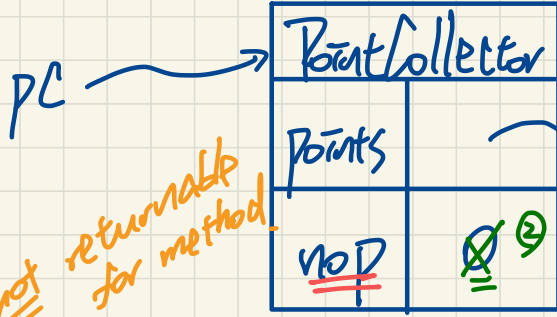
```
    pc.addPoint(-3, -4);  
    assertEquals(3, pc.getNumberOfPoints());
```

```
    pc.addPoint(3, -4);  
    assertEquals(4, pc.getNumberOfPoints());
```

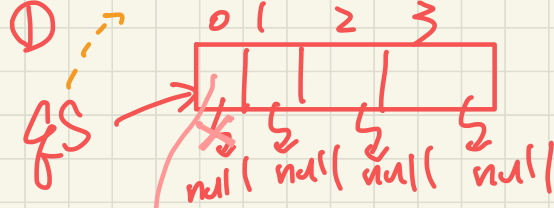
```
    Point[] ps = pc.getPointsInQuadrantI();  
    assertEquals(1, ps.length);  
    assertTrue(ps[0].getX() == 3 && ps[0].getY() == 4);  
}
```



find the left-most "null slot" in the array.



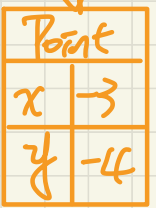
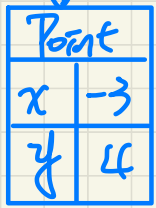
not returnable for method.



~~0~~ ~~2~~

~~4~~ ~~4~~ 2

4



Count of I

gl points

