```
// Lab 6: Polygon
 1
 2
    // I had to turn this one in late. :(
 3
    // Andrea Smith
 4
    // CSCI 1913
 5
 6
   class Polygon
7
 8
      private int[] sideLengths;
 9
      public Polygon(int sides, int lengths)
10
11
        int index = 0;
12
13
        sideLengths = new int[sides];
        for (int length: lengths)
14
15
        {
          sideLengths[index] = length;
16
          index += 1;
17
        }
18
19
      }
20
      public int side(int number)
21
22
      {
23
        return sideLengths[number];
24
      }
25
26
      public int perimeter()
27
28
        int total = 0;
29
        for (int index = 0; index < sideLengths.length; index += 1)</pre>
        {
30
          total += side(index);
31
        }
32
33
        return total;
34
      }
35
    }
36
    // Begin lab
37
38
    class Rectangle extends Polygon
39
    {
40
      int width = 0;
      int length = 0;
41
42
43
      public Rectangle(int width, int length)
44
        super(4, width, length, width, length);
45
        this.width = width;
46
        this.length = length:
47
```

```
48
      }
49
50
      public int area()
51
52
        return width*length;
53
54
55
    // perimeter is inherited
    }
56
57
58
59
    class Square extends Rectangle
60
61
62
      private int length;
63
64
      public Square(int length)
65
        super(length, length);
66
        this.length = length;
67
68
      }
69
70
      public int area()
71
        return length*length;
72
73
74
75
    // perimeter is inherited here also
    }
76
77
    // SHAPES. Public tests for the classes RECTANGLE and SOUARE.
78
    Comments show
79
    // what each test must print, and how many points it is worth.
80
    class Shapes
81
    {
82
      public static void main(String[] args)
83
84
        Rectangle wreck = new Rectangle(3, 5);
85
86
87
        System.out.println(wreck.side(0));
                                                 // 3
                                                          1 point.
                                                          1 point.
88
        System.out.println(wreck.side(1));
                                                 //
                                                     5
        System.out.println(wreck.side(2));
                                                 // 3
                                                          1 point.
89
        System.out.println(wreck.side(3));
                                                 // 5
                                                          1 point.
90
        System.out.println(wreck.area());
91
                                                 // 15
                                                         1 point.
        System.out.println(wreck.perimeter());
92
                                                 // 16
                                                         1 point.
```

 \cap

```
95
         Square nerd = new Square(7);
 94
 95
96
         System.out.println(nerd.side(0));
                                                         1 point.
                                                 //
                                                     7
 97
         System.out.println(nerd.side(1));
                                                 //
                                                     7
                                                         1 point.
         System.out.println(nerd.side(2));
                                                 //
                                                         1 point.
 98
                                                     7
         System.out.println(nerd.side(3));
                                                 //
                                                         1 point.
 99
                                                     7
                                                         1 point.
100
         System.out.println(nerd.area());
                                                 //
                                                     49
         System.out.println(nerd.perimeter());
                                                 //
101
                                                     28
                                                         1 point.
102
       }
103
     }
104
```