Consider this drawing:

We want to represent this as a recursive structure. To do that, you need to identify pieces of this drawing that have as components smaller versions of themselves. You also need to identify a base case, that is, a piece of the drawing that does not have smaller parts, but instead ends the drawing.

As with the Scribbler example, we want to define an interface with two implementing classes. One class is for the base case, while the other is for the part that recurses.
To make it a bit easier to see what is going on, here is another version that does not recurse as much. I have also changed the color on the ends so they are easier to see.

What part of the drawing corresponds to the base class?

What part of the drawing corresponds to the recursive part?

Here is an interface that you could use in this program.

```java
/**
 * BroccoliPart is an interface for pieces of broccoli.
 */
public interface BroccoliPart {

    /**
     * Draw the broccoli part recursively
     * @param g the graphics to draw on
     */
    public void paint(Graphics2D g);

    /**
     * Move the broccoli part recursively
     * @param x distance to move horizontally
     * @param y distance to move vertically
     */
    public void move(double x, double y);
}
```

What instance variables do you need in your recursive class to keep track of the pieces?
How would you decide when to use the base class?

How would you define the move method in your recursive class?