

Read over the definition below. I've omitted any check-expect expressions because I want you to predict what you expect:

```
(require picturing-programs)

; g : number -> image
(define (g d)
  (cond
    [(> d 0)
     (overlay
      (beside (g (- d 1)) (g (- d 1)))
      (square (* 2 (image-width (g (- d 1)))) "outline" "green"))]
    [else (square 10 "outline" "green")]))
```

Without running the code in DrRacket, predict what `(g 0)` produces. To do this, read over the code, replacing each occurrence of the placeholder "d" by 0.

Still without the code in DrRacket, predict what `(g 1)` produces. Again read over the code, replacing each occurrence of the placeholder "d" by 1. At this point you already know what `(g 0)` produces.

Now predict what `(g 2)` produces. As before, read over the code, replacing each occurrence of the placeholder "d" by 2. And, you already know what `(g 1)` produces.

Now read over this code. Again, I have omitted any check-expect expressions.

```
(require picturing-programs)

; k : number -> image
(define (k d)
  (cond
    [(> d 0)
     (beside/align
      "bottom"
      (k (- d 1))
      (rotate 60 (k (- d 1)))
      (rotate -60 (k (- d 1)))
      (k (- d 1)))]
    [else (line 5 0 "blue")]))
```

Predict what (k 0) produces. You should replace all occurrences of the placeholder "d" in the definition by 0, and then carefully trace the code.

Predict what (k 1) produces. Replace all occurrences of the placeholder "d" in the definition by 1, and then trace the code. At this point you already know what (k 0) produces.

Predict what (k 2) produces. Replace all occurrences of the placeholder "d" in the definition by 2, then trace the code. At this point you already know what (k 1) produces.

Finally, read over this code, which has been shorn of all its check-expect expressions:

```
(require picturing-programs)

; s : number -> image
(define (s d)
  (cond
    [(> d 0)
     (beside/align "bottom"
                   (line 0 (+ 5 (* 2 d)) "red")
                   (rotate 90 (s (- d 1))))]
    [else (line 0 5 "red")]))
```

Predict what (s 0) produces. To do this, replace all occurrences of the placeholder "d" by 0, and then trace the code.

Predict what (s 1) produces. As usual, replace all occurrences of the placeholder "d" by 1, and then trace the code. You already know what (s 0) produces.

Predict what (s 2) produces. Replace all occurrences of the placeholder "d" by 2, and trace the code. You already know what (s 1) produces.