

CSSE 220

Recursion

Checkout *InClassRecursion* project from SVN

Recursion

- A solution technique where the same computation **occurs repeatedly** as the problem is solved



recurs

- Examples:
 - Sierpinski's Triangle
 - Towers of Hanoi:
<http://www.mathsisfun.com/games/towerofhanoi.html>
or search for Towers of Hanoi

An example – Triangle Numbers

- If each red block has area 1, what is the **area $A(n)$** of the Triangle whose *width* is n ?

– Answer:

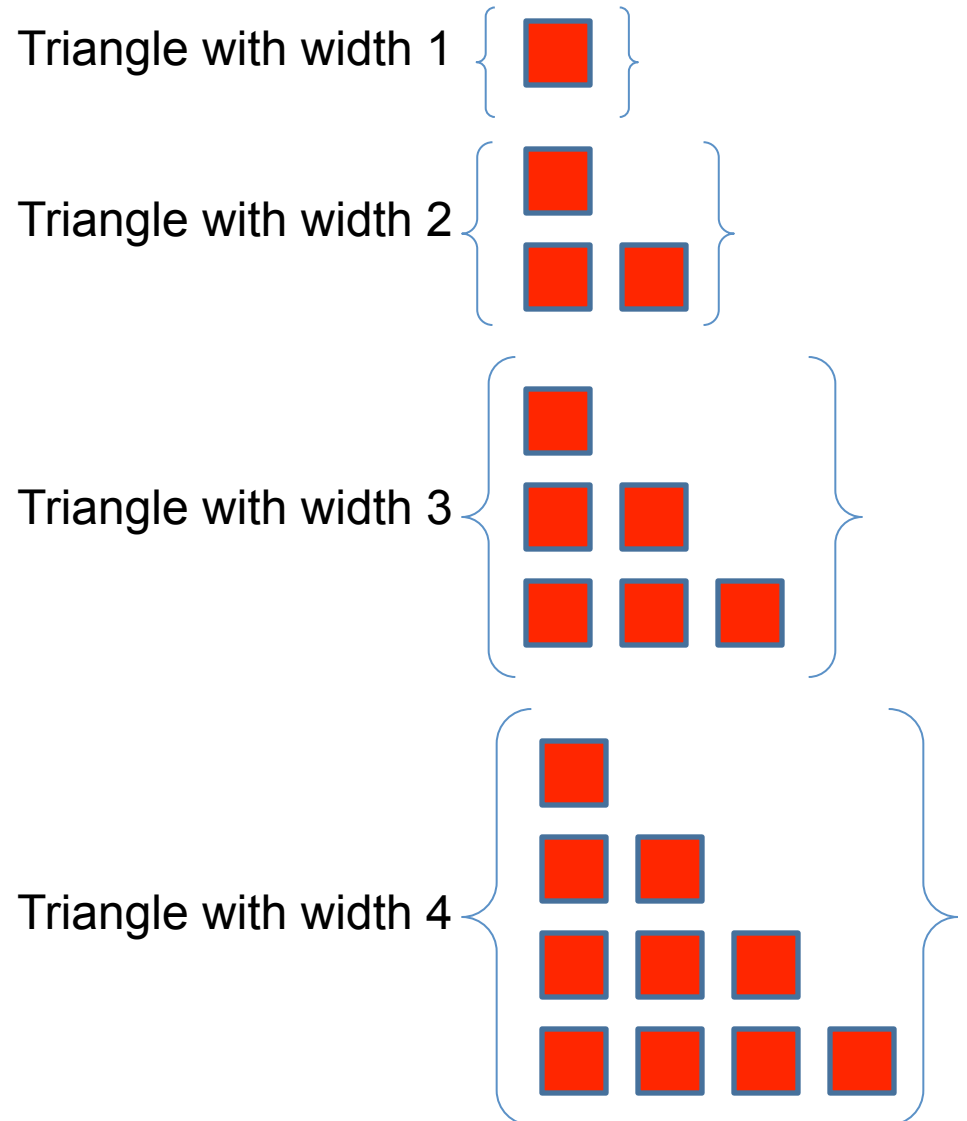
$$A(n) = n + A(n-1)$$

- The above holds for which n ? What is the answer for other n ?

– Answer: The recursive equation holds for

$$n \geq 1.$$

For $n = 0$, the area is 0.



Frames for Tracing Recursive Code

1. Draw box when method starts

2. Fill in method name and arguments.

method name(values passed in)

Parameters = parameter values

Base case check (Yes – return) OR (No – continue)

local variables = values (when/if declared)

Return statement

ret val

4. Show the check of the base case and result, continue or return?

3. List every parameter and its argument value.

5. List every local variable declared in the method, **update values when set**

Thanks to David Gries for this technique

6. Step through the method, update the variable values, add a “return value Box” and draw new frame for new calls

Examples

- Let's look at the first two examples in the code and on the quiz

Key Rules to Using Recursion

- ▶ Always have a **base case** that **doesn't recurse**
- ▶ Make sure recursive case always **makes progress**, by **solving a smaller problem**
- ▶ **You gotta believe**
 - Trust in the recursive solution
 - Just consider one step at a time

Programming Problem

- Let's look at SimplePalindrome. There are a couple ways to approach this

Recursive Helpers

- ▶ Our `isPalindrome()` makes lots of new String objects (using `substring`)
- ▶ We can make it better with a “recursive helper method”
 - ▶ Many recursive problems require a helper method

```
public boolean isPalindrome() {  
    return isPalindrome(0, this.text.length() - 1);  
}
```

Position of first letter of the remaining String to check

Position of last letter of the remaining String to check

And the rest...

- Let's look at the remaining examples in the quiz and in eclipse.

Practice Practice Practice

- Head to <http://codingbat.com/java/Recursion-1> and solve 5 problems. I personally like bunnyEars, bunnyEars2, count7, fibonacci, and noX
- Get help from me if you get stuck
- Then take a look at the recursion homework