CSSE 220

Recursion

Checkout *Recursion* project from SVN

Recursion

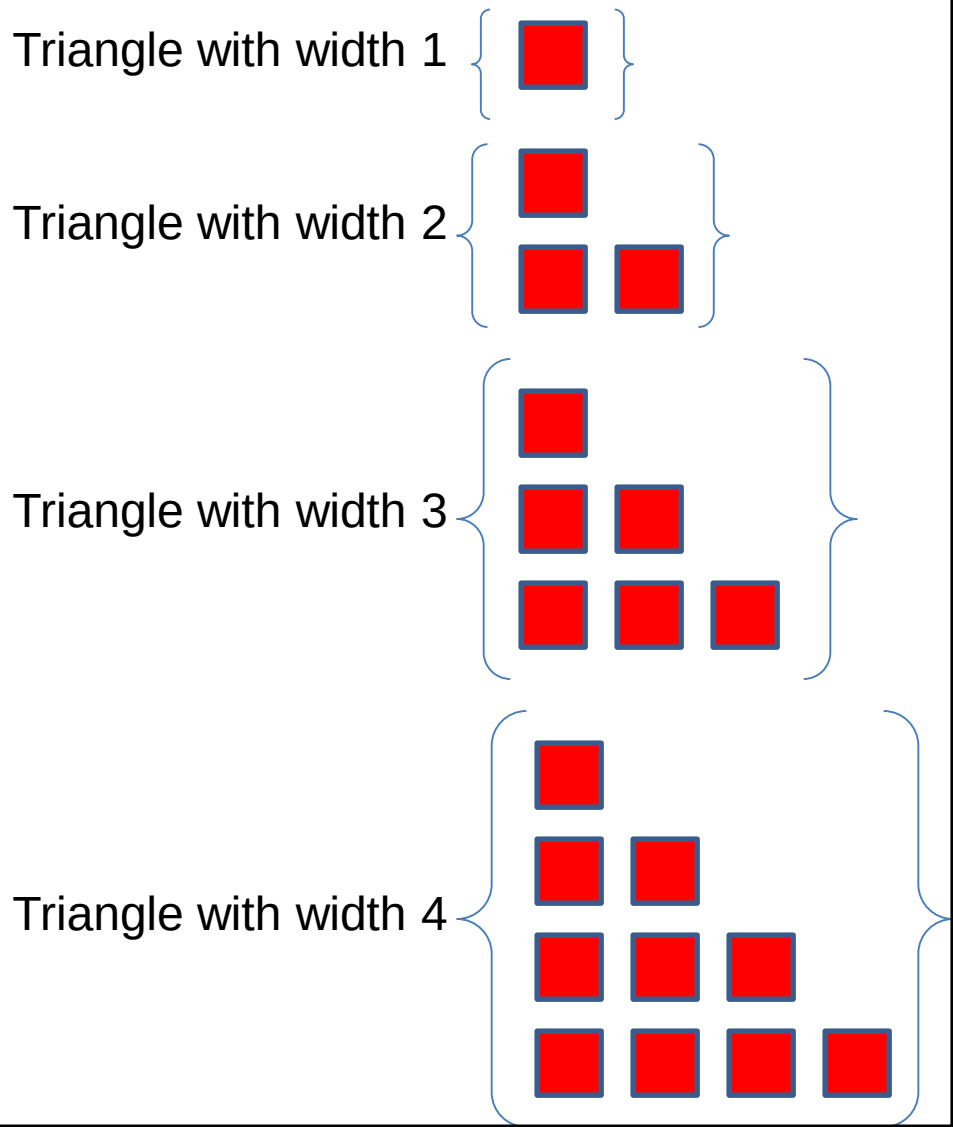
- A solution technique where the same computation **occurs repeatedly** as the problem is solved

- Examples:
  - Sierpinski Triangle: https://en.wikipedia.org/wiki/Sierpinski_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 >= 1.
    For n = 0, the area is 0.



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

Frames for Tracing Recursive Code

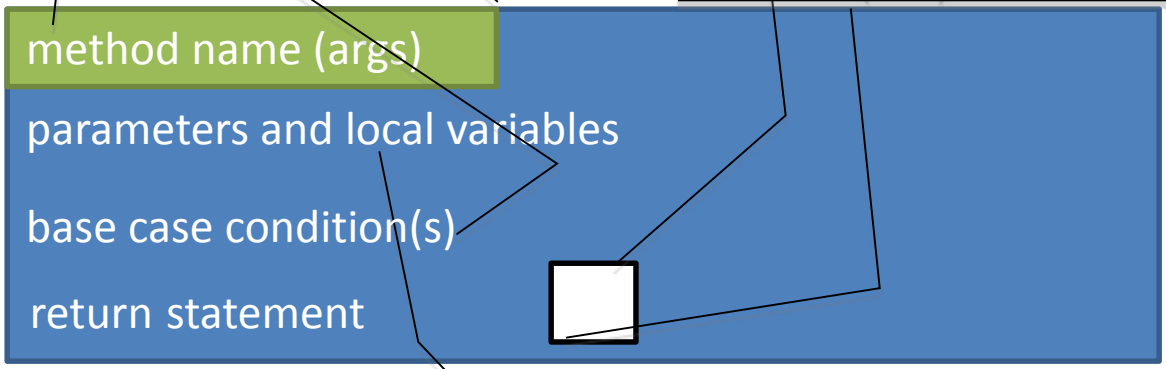
1. Draw box when method starts

2. Fill in name

5. Check Condition(s)

6. Add box for next recursive call frame. Add blank for unknown value

7. Add blank for unknown value, if needed (may be box from #6)



3. List every parameter and its argument value.

4. List every local variable declared in the method, **but no values yet**

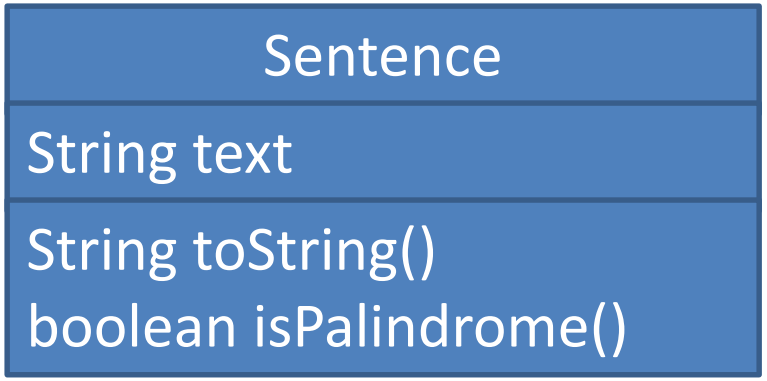
8. Step through the method, update variable values, draw new frame for new calls

Thanks to David Gries for this technique

Q1-Q2

Programming Problem

- Add a recursive method to Sentence for computing whether Sentence is a palindrome



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