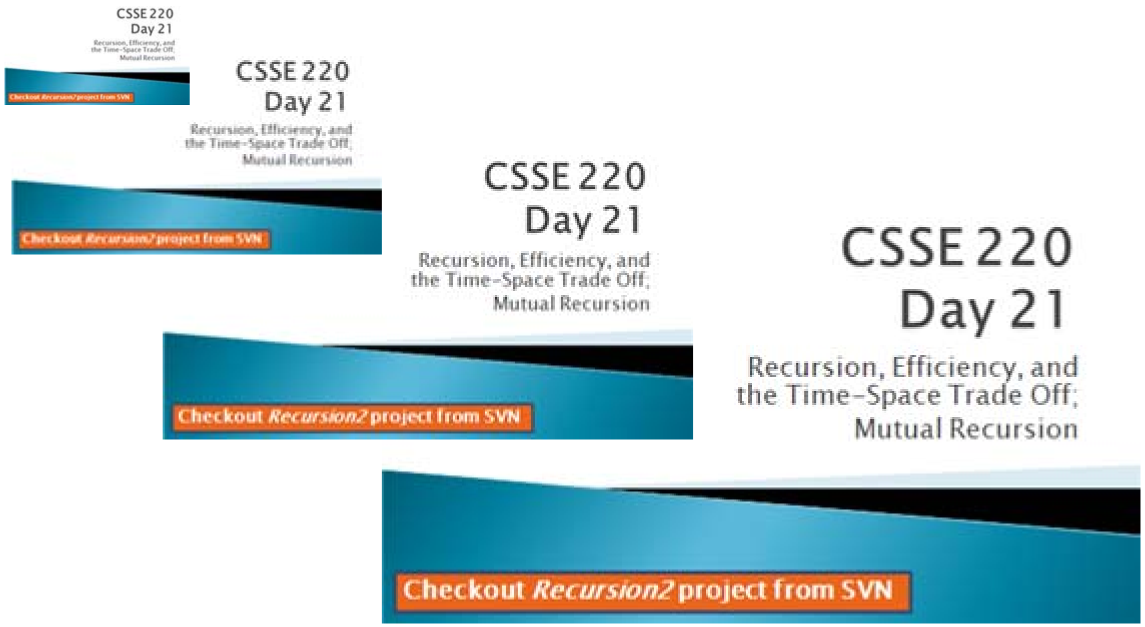

CSSE 220 Day 21
Recursion, Efficiency, and the Time-Space Trade Off; Mutual Recursion
Checkout *Recursion2* project from SVN

CSSE 220 Day 21
Recursion, Efficiency, and the Time-Space Trade Off; Mutual Recursion
Checkout *Recursion2* project from SVN

CSSE 220 Day 21
Recursion, Efficiency, and the Time-Space Trade Off; Mutual Recursion
Checkout *Recursion2* project from SVN

CSSE 220 Day 21
Recursion, Efficiency, and the Time-Space Trade Off; Mutual Recursion
Checkout *Recursion2* project from SVN

CSSE 220 Day 21

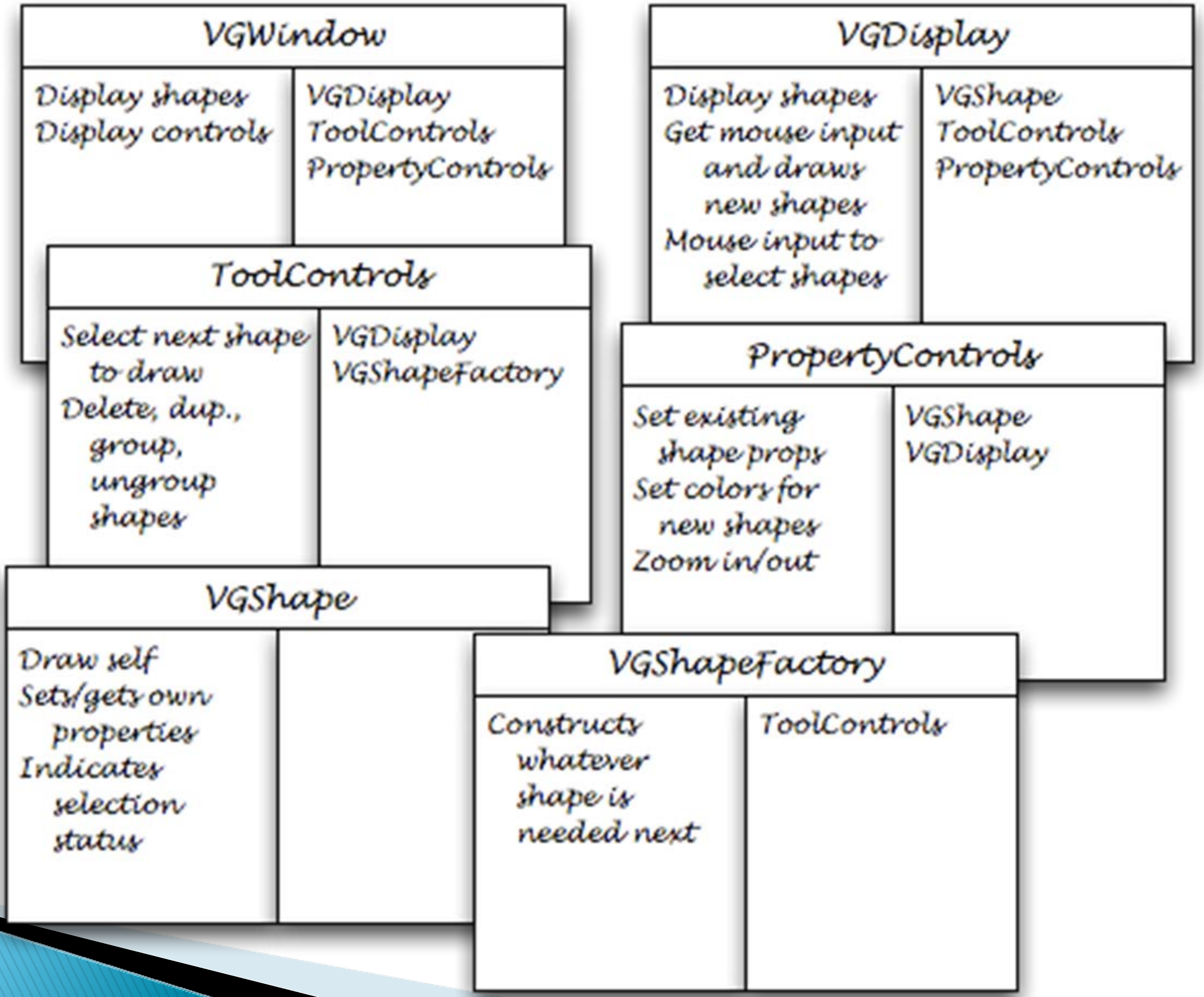
Recursion, Efficiency, and the Time-Space Trade Off; Mutual Recursion

Checkout *Recursion2* project from SVN

No late day for VectorGraphics

- Why? Because the exam is 24 hours after the due time, and you ought to focus on the exam rather than the project at that point.

Sample CRC Cards



VGWindow	
Display shapes	VGDisplay
Display controls	ToolControls
	PropertyControls

VGDisplay	
Display shapes	VGShape
Get mouse input and draws new shapes	ToolControls
Mouse input to select shapes	PropertyControls

ToolControls	
Select next shape to draw	VGDisplay
Delete, dup., group, ungroup shapes	VGShapeFactory

PropertyControls	
Set existing shape props	VGShape
Set colors for new shapes	VGDisplay
Zoom in/out	

VGShape	
Draw self	
Sets/gets own properties	
Indicates selection status	

VGShapeFactory	
Constructs whatever shape is needed next	ToolControls

Session 20 Leftovers

Recursive Functions

- Factorial:

Base Case

$$n! = \begin{cases} 1 & \text{if } n \leq 1 \\ n * (n-1)! & \text{otherwise} \end{cases}$$

Recursive step

- Ackermann function:

$$A(m,n) = \begin{cases} n+1 & \text{if } m = 0 \\ A(m-1,1) & \text{if } m > 0 \text{ and } n = 0 \\ A(m-1, A(m, n-1)) & \text{otherwise} \end{cases}$$

Q1

Recursive Helpers

- Our isPalindrome() makes lots of new Sentence objects
- We can make it better with a “recursive helper method”
-
```
public boolean isPalindrome() {
    return isPalindrome(0, this.text.length() – 1);
}
```

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

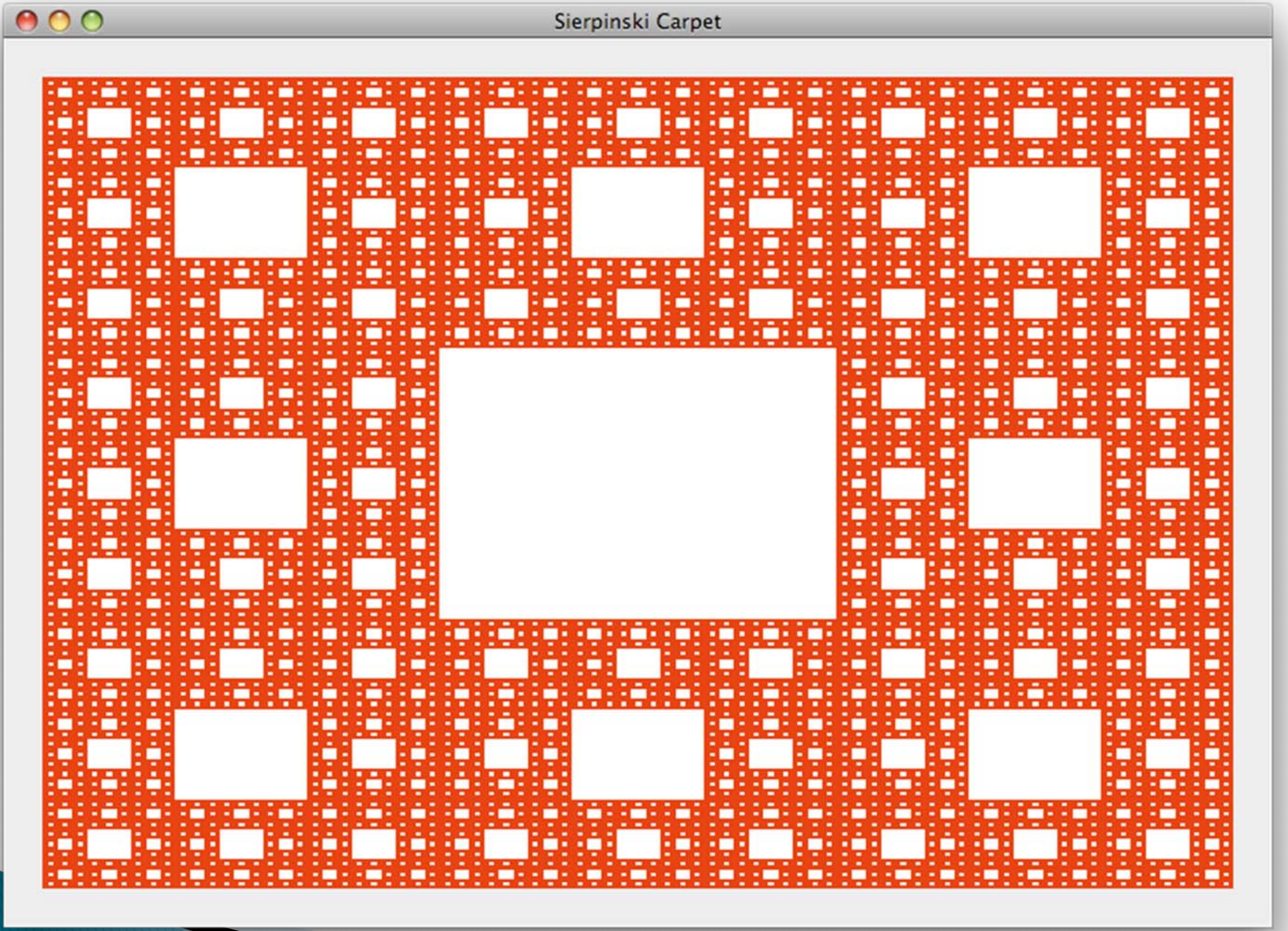
Another Definition of Recursion

- "If you already know what recursion is, just remember the answer. Otherwise, find someone who is standing closer to Douglas Hofstadter than you are; then ask him or her what recursion is."

  —Andrew Plotkin

Towers of Hanoi

- Demo
- Code (on the board)
- Analysis

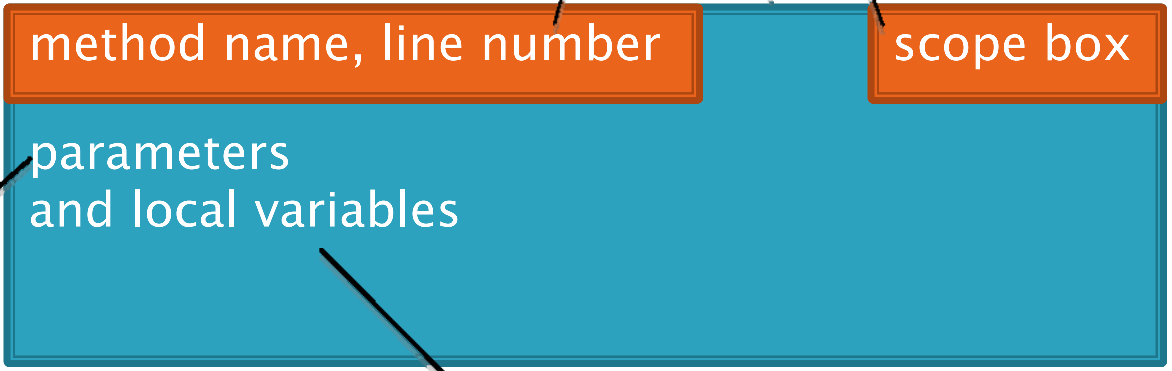

Sierpinski Carpet

Frames for Tracing Recursive Code

1. Draw box when method starts

2. Fill in name and first line no.

3. Write class name (for static method) or draw reference to object (for non-static method)



4. List every parameter and its argument value.

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

6. Step through the method, update the line number and variable values, draw new frame for new calls

7. “Erase” the frame when the method is done.

Thanks for David Gries for this technique

Q2-3

That's Slow!

- Why does recursive Fibonacci take so long?!?

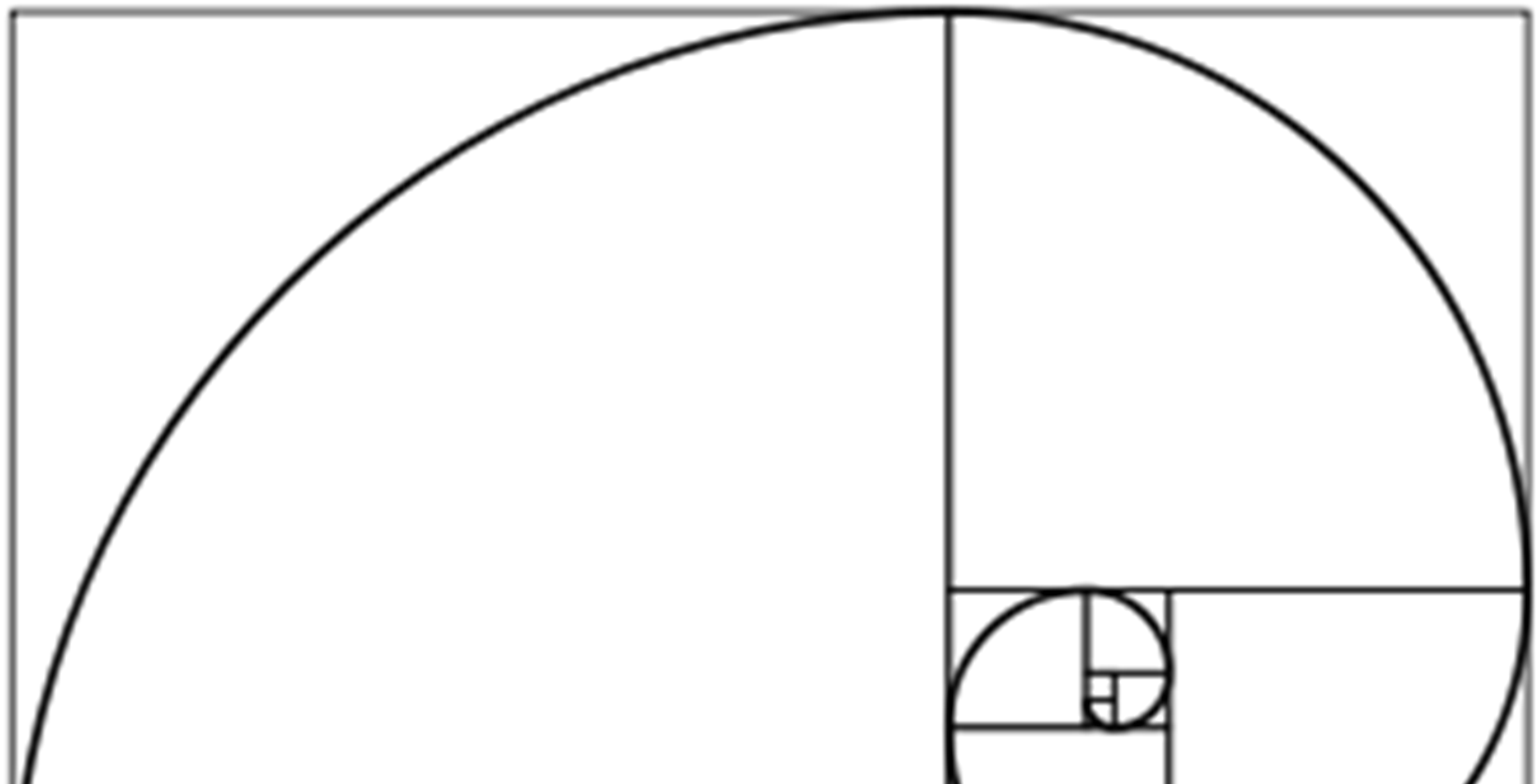
- Can we fix it?

Classic Time-Space Trade Off

- A deep discovery of computer science
- In a wide variety of problems we can tune the solution by varying the amount of storage space used and the amount of computation performed
- Studied by “Complexity Theorists”
- Used everyday by software engineers

Q4

Fibonacci Sequence in Squares



Mutual Recursion

- Two or more methods that call each other repeated

Q5

Example

- Hofstadter Female and Male Sequences:

$$F(n) = \begin{cases} 1 & \text{if } n = 0 \\ n - M(F(n-1)) & \text{if } n > 0 \end{cases}$$

$$M(n) = \begin{cases} 0 & \text{if } n = 0 \\ n - F(M(n-1)) & \text{if } n > 0 \end{cases}$$

- Questions:
  - How often are the sequences different in the first 50 positions? first 500? first 5,000? first 5,000,000?

Vector Graphics Work Time

- Should have completed **Status Report** for Cycle 1 and listed **User Stories** for Cycle 2