

# CSSE 220 Day 23

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

# CSSE 220 Day 23

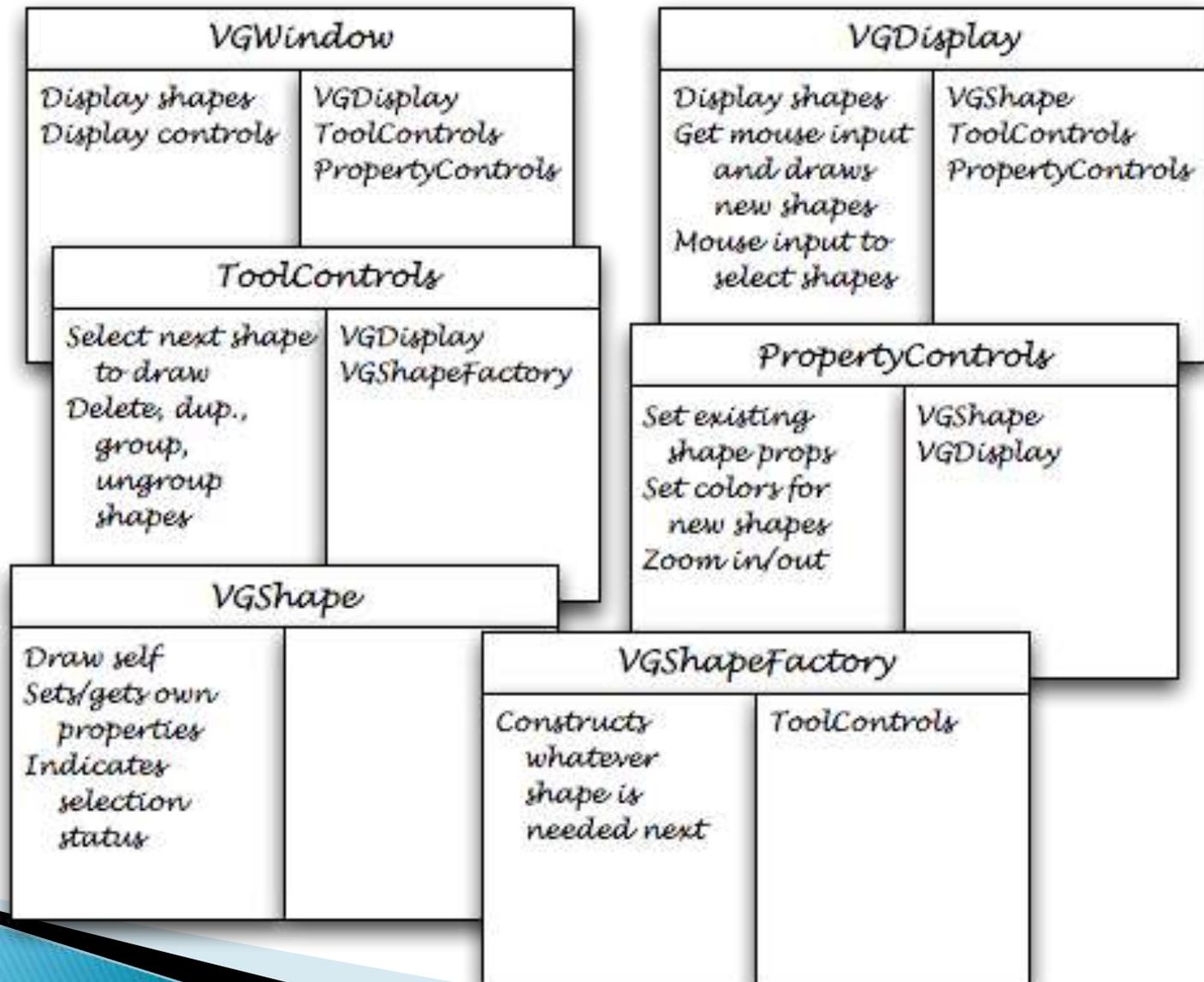
Recursion, Efficiency, and  
the Time-Space Trade Off;  
Mutual Recursion

Checkout *Recursion2* project from SVN

# Exam Review

- ▶ Exam 2, tomorrow, open-book, open notes
  - Can start at 7:30am if you want extra time
- ▶ Key Topics:
  - Cohesion
  - Coupling
  - Immutable Classes
  - Side Effects
  - `static`
  - Scope
  - Interfaces
  - Polymorphism
  - Event-based prog.:
    - `ActionListener`
    - `MouseListener`
  - Inheritance
  - Abstract classes
  - Object-oriented design
  - CRC Cards
  - Recursion

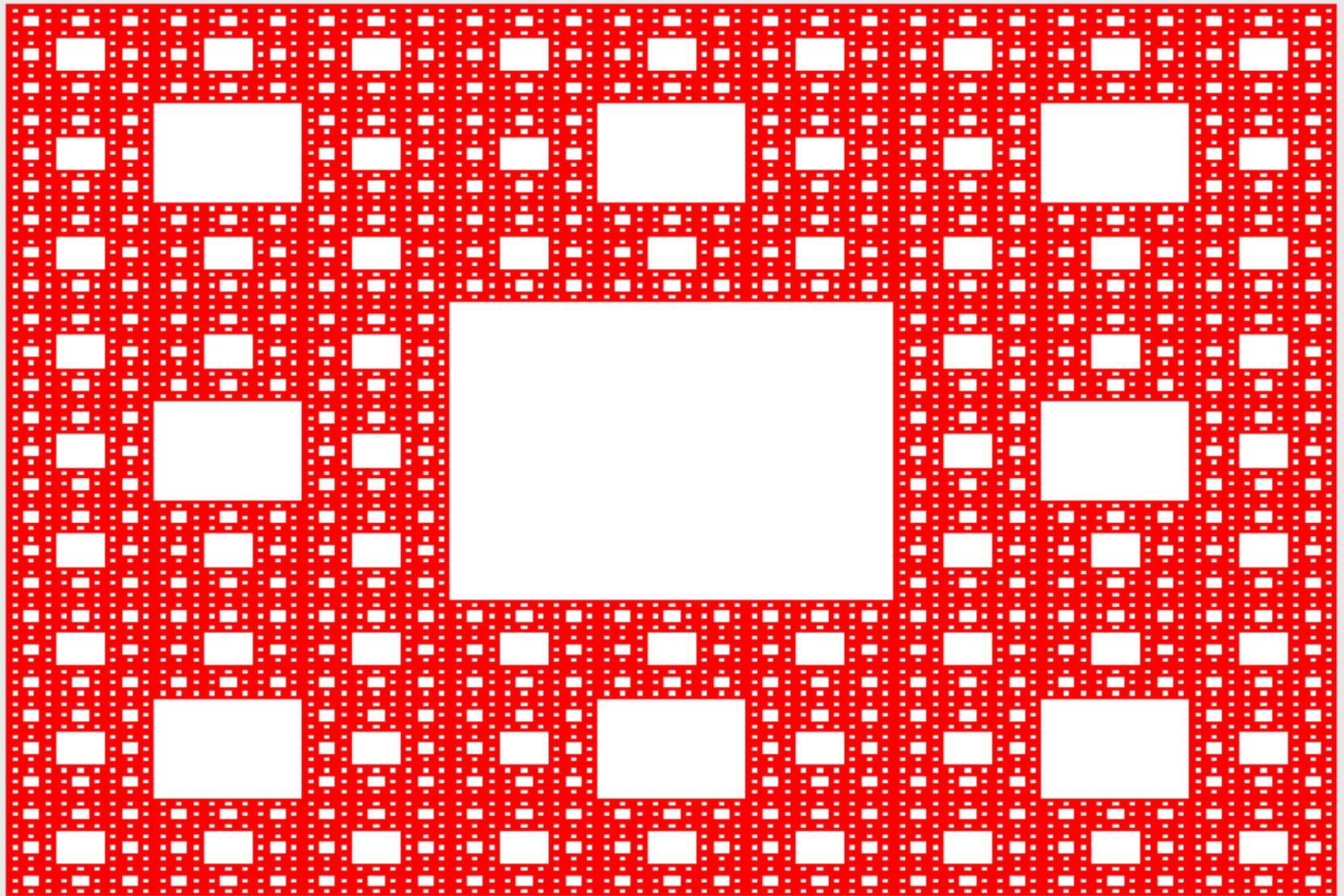
# Sample CRC Cards



# 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

Sierpinski Carpet

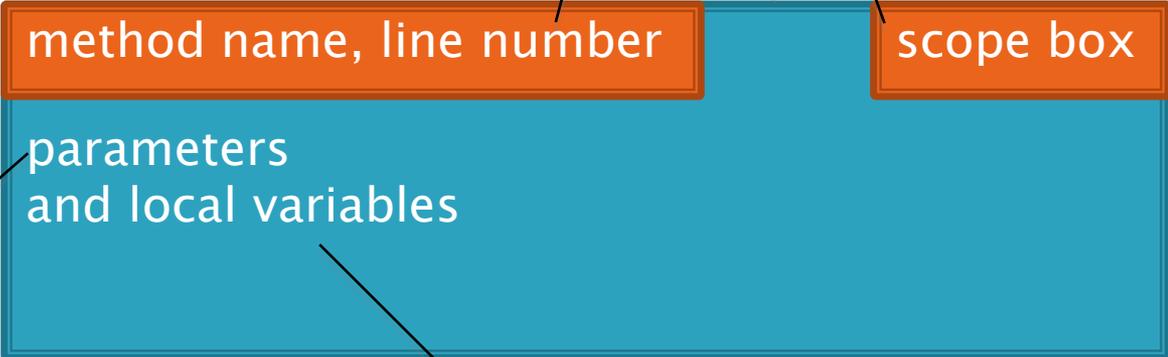


# Frames for Tracing Recursive Code

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

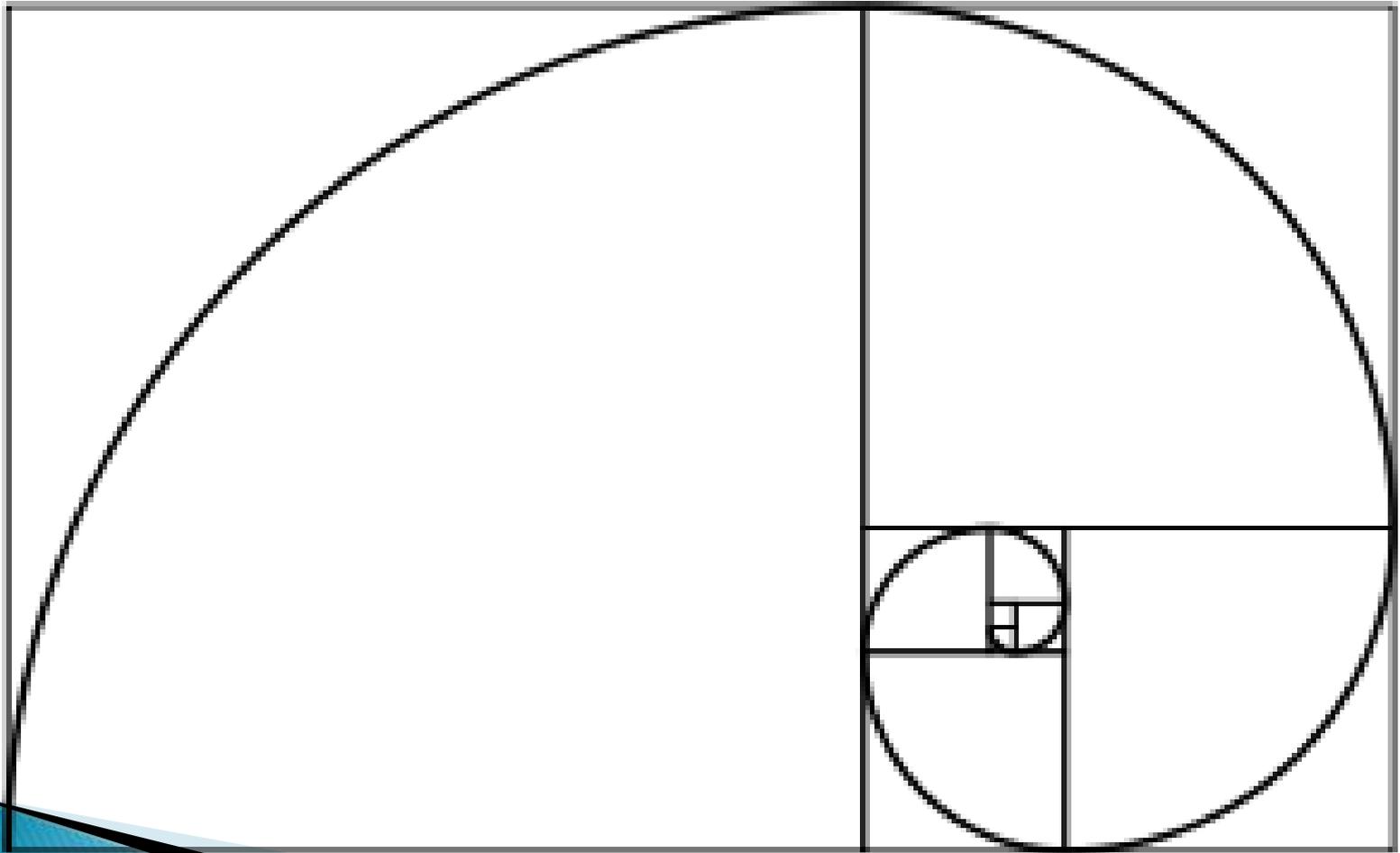
# What the Fib?

- ▶ 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

# Fibonacci Sequence in Squares



# Mutual Recursion

- ▶ Two or more methods that call each other repeated

# 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 5000?

# Vector Graphics Work Time

- »» Should have completed **Status Report** for Cycle 2 and listed **User Stories** for Cycle 3