Tracing Recursion

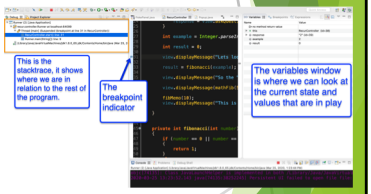
Cody Henrichsen

How to trace recursion

- What happens in the computer when a recursive method is called?
- How do we examine the process and look at the state?
- What should we be looking for?
- What does it look like when it is wrong?

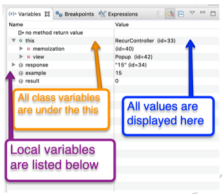
Debug

- These steps are listed for Eclipse but the process is similar in many other IDE's
- Add a breakpoint at the line of the recursive call
  - This is to tell the program to wait for the next debug instruction
- Start the debug process
- In the debug menu choose the option to "Step Into" the execution



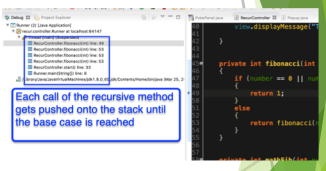
Look inside

- You want to look at the location in the stack trace to make sure you are at the right spot
- You will be in the first call to the recursive method and then you will examine the parameter to see if it meets the base case
- You can look at the current values in the Variables window
  - Class variables are in the this triangle
  - Local variables below

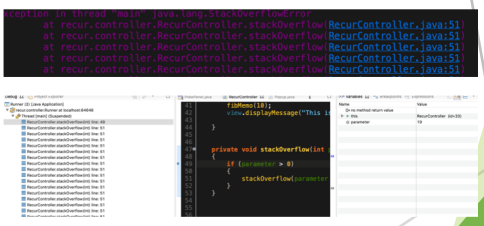


Recursive Call

- Each recursive call is placed on the stack
- Look in the stack trace to see where you are in recursion
- Great to match with this on a scratch paper to match mentally
- If you are not interested in the details remember to use step over to go to the next line
- Step return when you want to go to who called that method



What does it look like when it is WRONG




Review of terms
Step Into
Step Over
Step Return
Breakpoint
Resume
State
Value