Binary Trees

CS 62 - Spring 2016
Michael Bannister

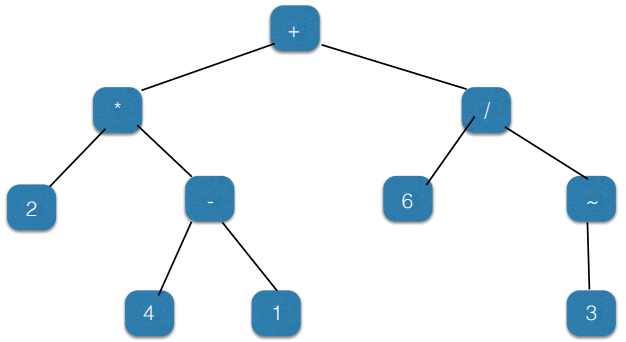
First Exam on Friday!

This Week's Assignments

- Lab:
  - Debugger: Inspect memory (including run-time stack and heap) to see what is happening in program.
- Assignment:
  - Postfix calculator

Evaluate Expression Tree

- Evaluate left subtree, right subtree, perform operation at root.



Animals Game

- Guess animal using only true-false questions.
- See demo program

Look at BinaryTree.java

Notice leaves are nodes w/null values

Tree Traversals

- Traversals:
  - Pre-Order: root, left subtree, right subtree
  - In-Order: left subtree, root, right subtree
  - Post-Order: left subtree, right subtree, root
- Most algorithms have two parts:
  - Build tree
  - Traverse tree, performing operations on nodes

Recursive In-order

```
if (!isEmpty()){
  left.inOrder()
  doSomething to this.value()
  right.inOrder()
}
```

Types of Iterators

- Pre-order: root, left subtree, right subtree
- Post-order: left subtree, right subtree, root
- In-order: left subtree, root, right subtree.

Stack Based Iterators

- Uses a stack to simulate the call stack from recursive implementation
- Each stack “frame” needs to record current line number and current node.
- Example on board.