


Lecture 15: Binary Trees

The graphic consists of a large blue square on the left containing a small white plus sign. To its right are two columns of two smaller squares each. The top row contains an orange square and a green square. The bottom row contains a purple square and a red square.

+ Today



- Reading
 - JS 12.1-12.5
- Objectives
 - Binary Trees
- Announcements
 - Midterm quiz will be March 10th

+ Programming assignment

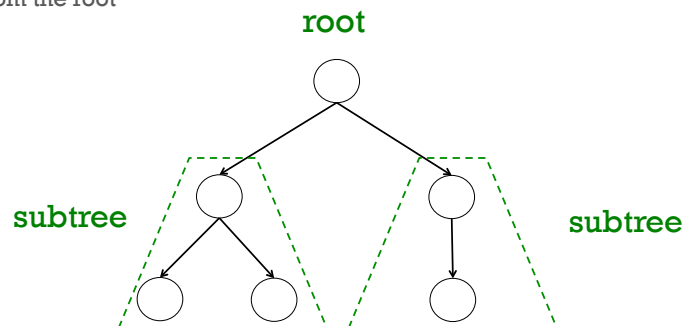
- Calculator
 - Practice using Stacks

- Class structure
 - Calculator
 - State
 - DigitButtonListener
 - OpButtonListener



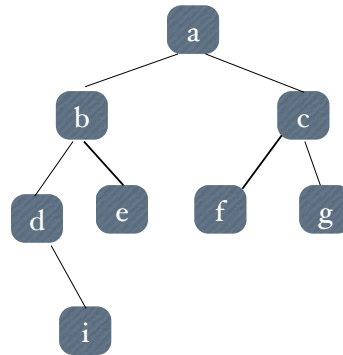
+ Tree Definition

- A tree is either
 - empty or
 - consists of a node (called a root node) together with a collection of trees (called subtrees). The subtrees are disjoint from each other and from the root



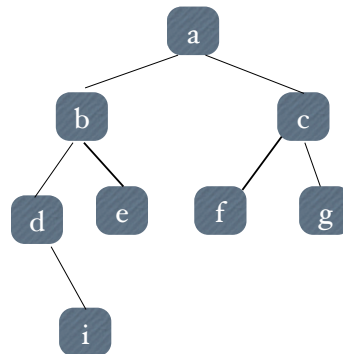
+ Tree Terminology

- edge – connects a node to its subtree
- parent/child – a parent node is directly above a child node
- siblings – nodes that have the same parent
- ancestors/predecessors – the ancestors of n are n , n 's parents, n 's parent's parent, etc.



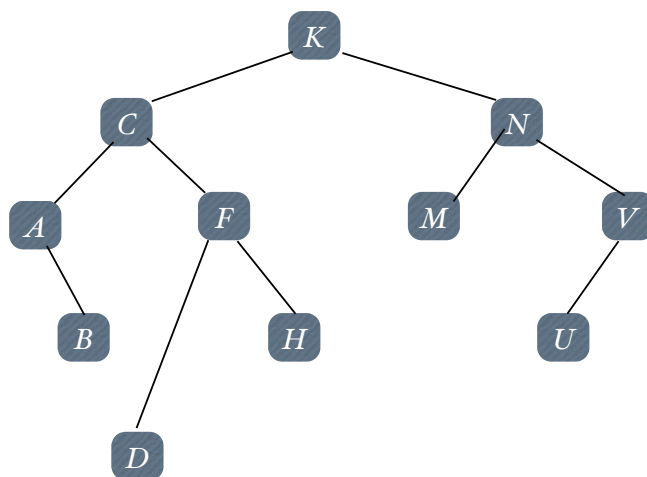
+ Tree Terminology

- descendants/successors – the descendants of n are n , n 's children, n 's children's children, etc.
- leaf/interior node – leaves have no children. Any node that is not a leaf is an interior node
- degree – number of children of a node
- forest – a set of trees



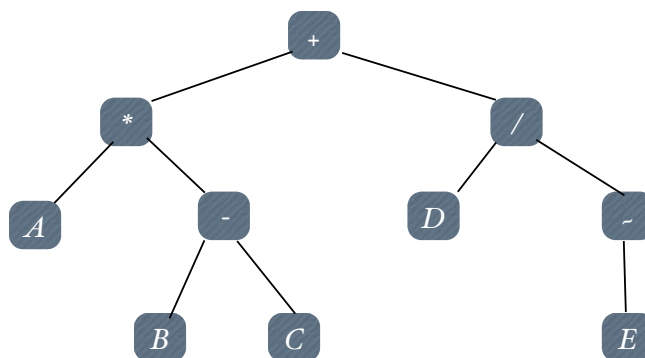
+ Example: Binary Search Tree

K, C, A, N, B, V, F, U, D, H, M

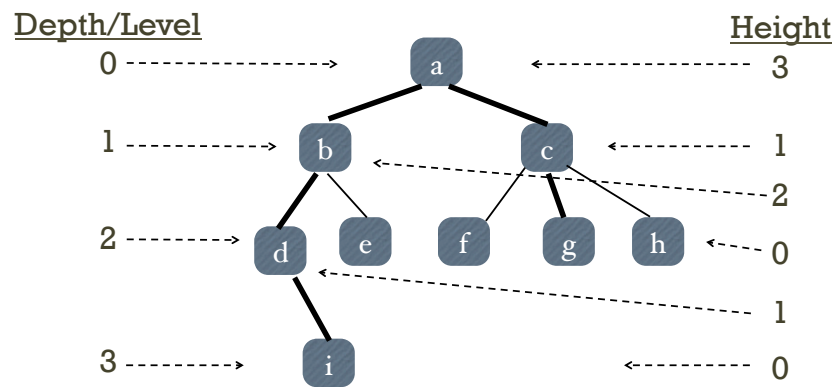


+ Example: Expression Tree

$[A * (B - C)] + (D / \sim E)$



+ Path Terminology (on board)



+ Binary Tree Theorems (on board)

