Today

- Balanced BSTs

Assignments

- HW 7 out
- Quiz 9 on Thurs
Review: Full, Complete, and Balanced Trees

**Full:** (i) internal nodes have 2 children, (ii) ea. node’s subtrees same height

**Complete:** full at level $n - 1$, level $n$ filled from left to right

**Balanced:** each node’s left and right subtree heights differ by at most 1
AVL Trees ... (Adelson-Velsky and Landis)

Basic Idea: (animated: www.cs.usfca.edu/~galles/visualization/AVLtree.html)
- Calculate a “balance factor” for each node (left subtree – right subtree height)
  - Note: just store the height of ea. subtree root and calculate balance factor
- After every insert and delete, do “tree rotations” if needed
  - Either a single or double rotation may be needed
- Apply rotations (deletion) and update balance factors “up the tree”

Note: parent becomes 4’s right-child, and 4’s right-child becomes parent’s left-child
General case for single rotation (*left–left case*)

- Insertion in the *left* subtree of the *left* child
- Assumed that the tree is balanced before the rotation
- And becomes unbalanced after the rotation

Requires a "*right*" rotation:

![Diagram of single rotation (left-left case)](image)

After insert and rotation, done with rotations

- Subtree now rooted at \( k_1 \) is back to original height
- Since subtree is now balanced, no other rotations needed