

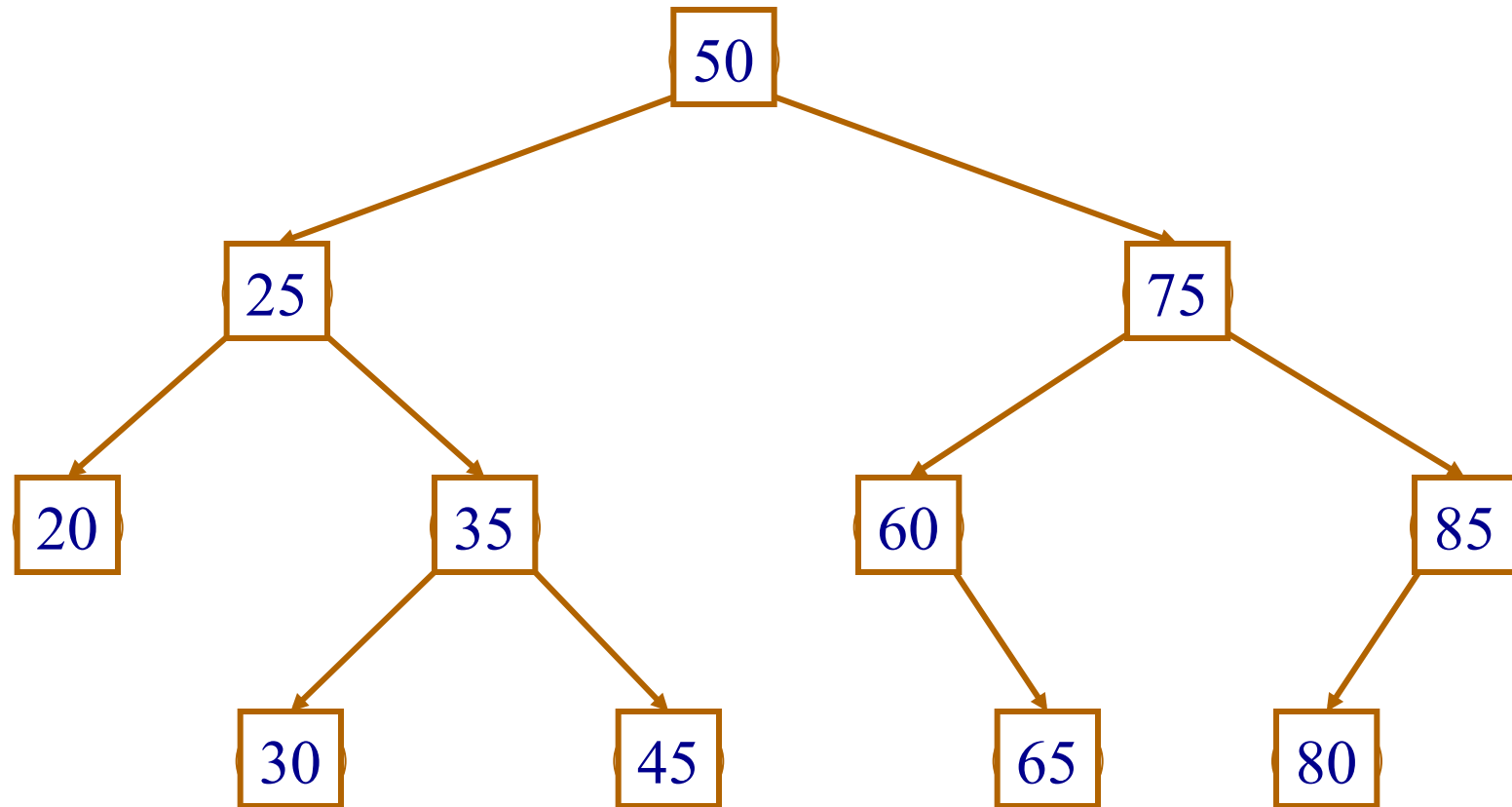
## AVL Trees

### Motivation and Introduction

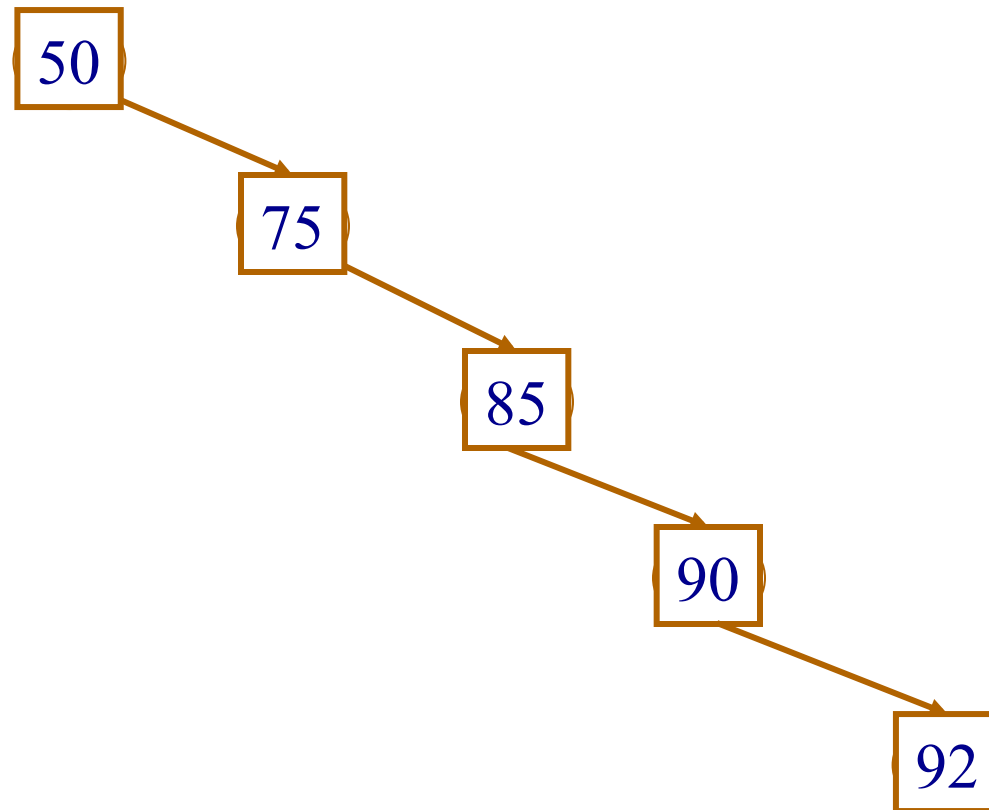
# Goals

- Pros/Cons of a BST
- AVL Solution
- Height-Balanced

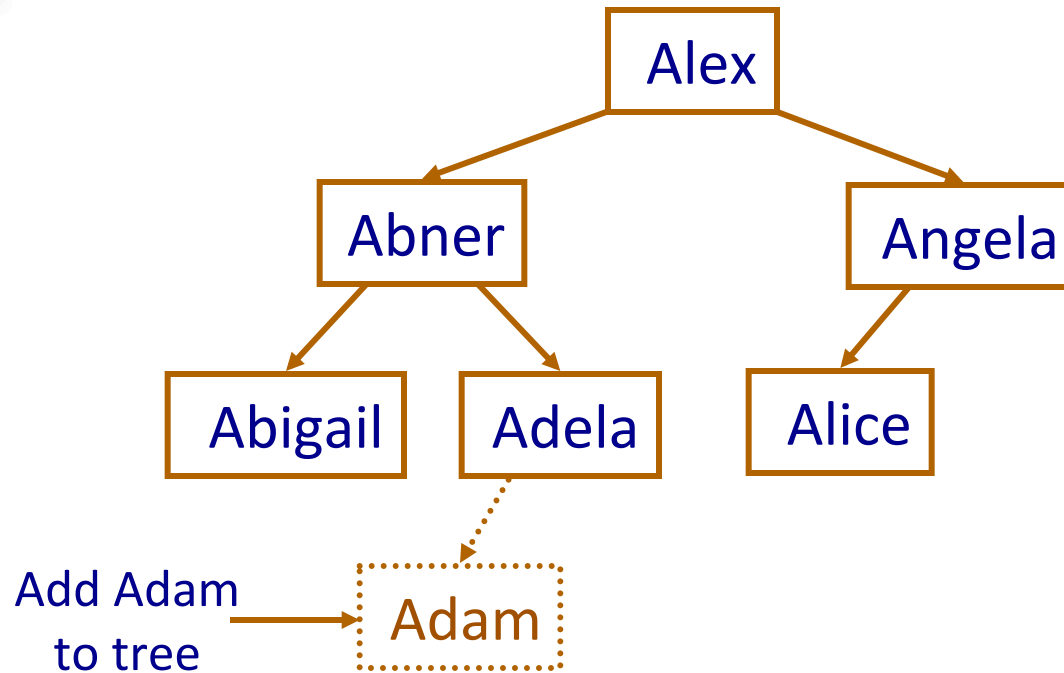
# Binary Search Tree: Balance



# Binary Search Tree: Balance



# Complete Binary Trees

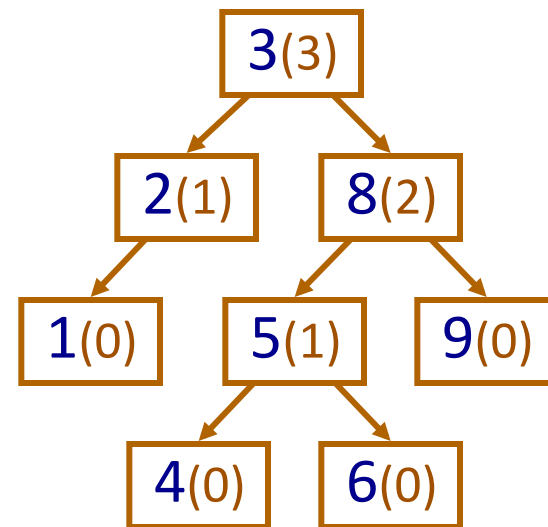


- Very costly to maintain a complete binary tree

# Height-Balanced BST

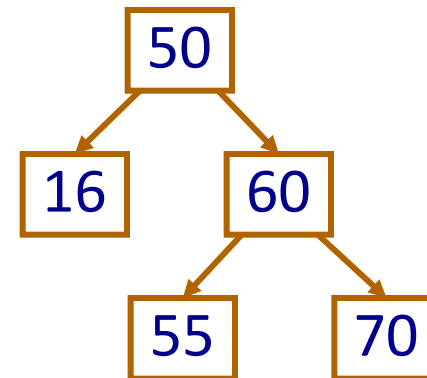
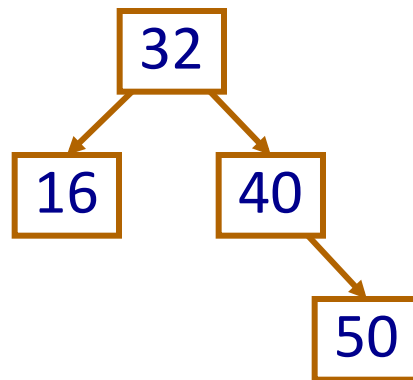
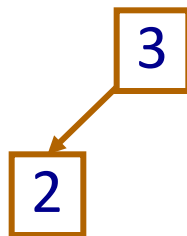
- For each node, the **height difference** between the left and right subtrees **is at most one**
- Trees are locally balanced, but globally they can be slightly more unbalanced

Height-Balanced Tree



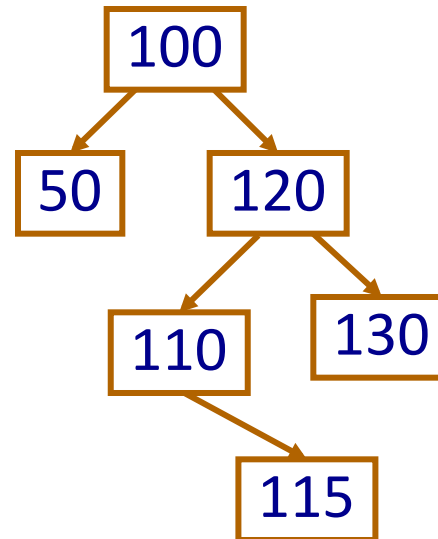
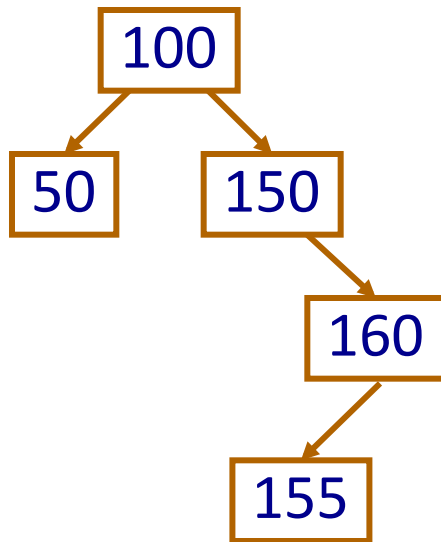
# Quiz

- Are these trees height balanced? If not, which node is out of balance?



# Quiz

- Are these trees height balanced? If not, which node is out of balance?



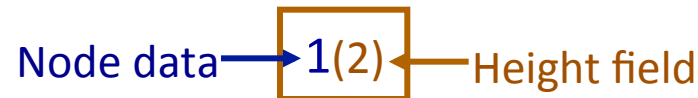


# Height-Balanced Trees

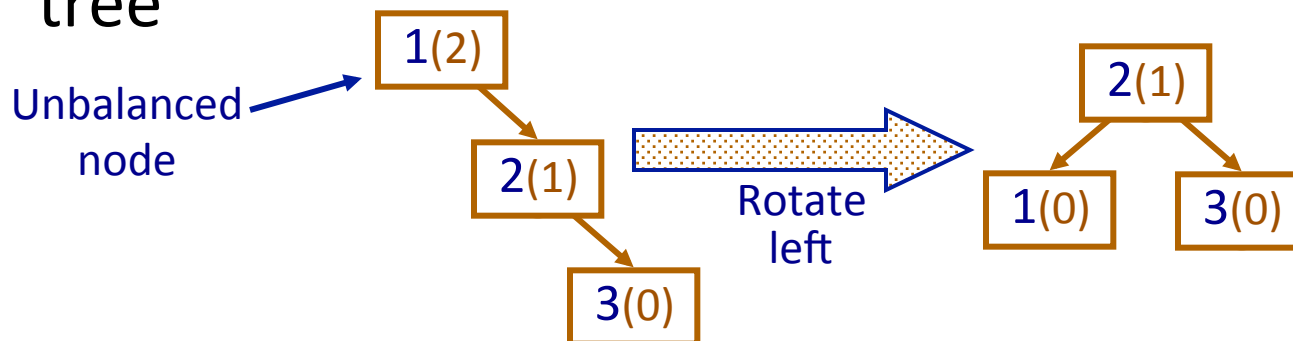
- Mathematically, the longest path in a height-balanced tree has been shown to be, at worst, 44% longer than  $\log n$
- Therefore, algorithms on height-balanced trees that run in time proportional to the path length are still  $O(\log n)$
- So.....How do we maintain height balance??

# AVL Trees

- Named after the inventors' initials: G.M. Adelson-Velskii, E.M. Landis
- Maintain the height balanced property of a BST through a series of rotations



- When unbalanced, performs a “rotation” to balance the tree



# Your Turn

- Read Chapter 10 and Worksheet 31
  - but do not yet work on the problems