AVL Trees
AVL Trees

• A “balanced” binary search tree
• AVL trees are balanced in the sense that for any node the height of the left subtree and the height of the right subtree differ by at most 1.
• To keep track of the balance a balance value (tall left, equal, tall right) can be attached to each node or each node can maintain its height.
AVL Trees

• The find functions for AVL trees work exactly like the find functions for binary search trees.
• Insert and delete follow the same procedure as that used by binary search trees except after a node is inserted or deleted the tree might have to be rebalanced.
• Rebalancing takes place along the search path.
• Inserts require rebalancing at most one node in the search path.
• Deletes could require rebalancing at every node in the search path.
AVL Single Rotation
Insertion Done to the Left of S

(a)

(b)
AVL Tree Double Rotation Insertion Done to the Left of S