CS 302 – AVL Tree Insertion

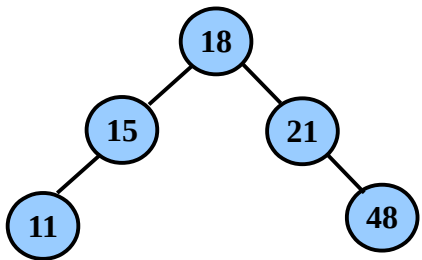Case 1: if (balance factor > 1 AND key < left node value) → right rotate
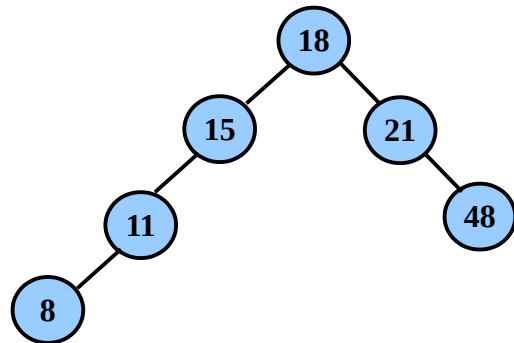Case 2: if (balance factor < -1 AND key > right node value) → left rotate
Case 3: if (balance factor > 1 AND key > left node value) → left node left rotate, right rotate
Case 4: if (balance factor < -1 AND key < right node value) → right node right rotate, left rotate
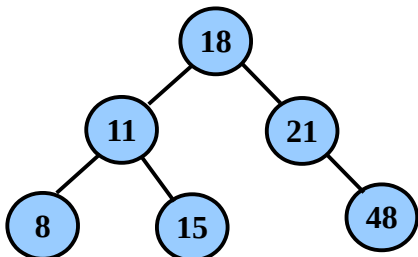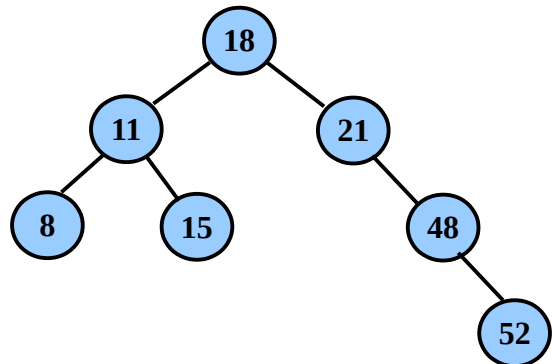
Insert: 18, 21, 15, 11, 48 (balanced)

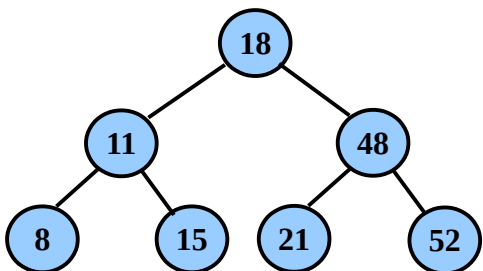Insert: 8 (step 1, unbalanced, case 1)
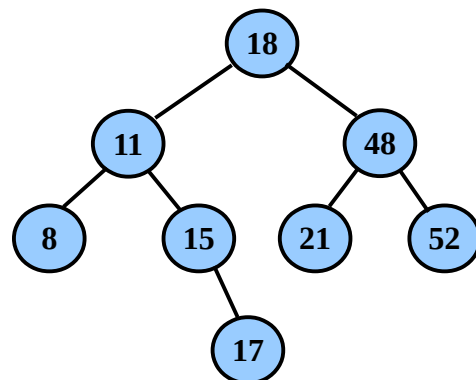
Insert: 8 (step 2, right rotate, balanced)
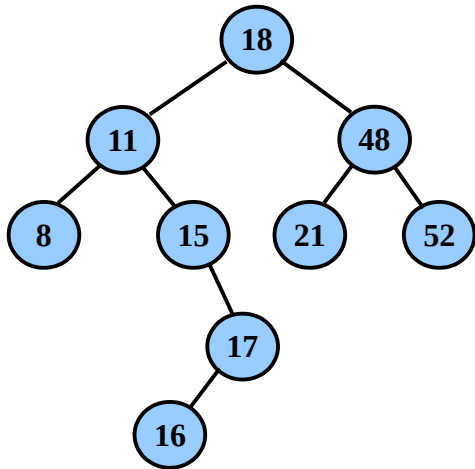
Insert: 52 (step 1, unbalanced, case 2)
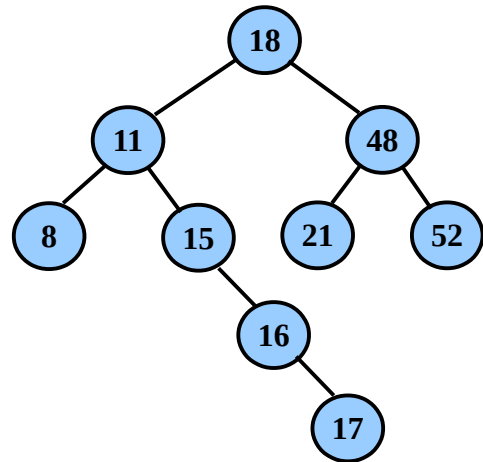
Insert: 52 (step 2, left rotate, balanced)
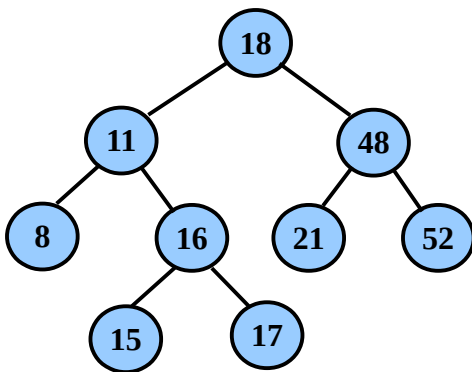
Insert: 17 (balanced)
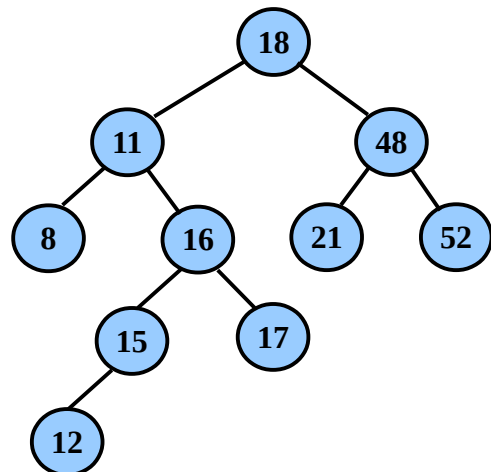
Insert: 16 (step 1, unbalanced, case 4)

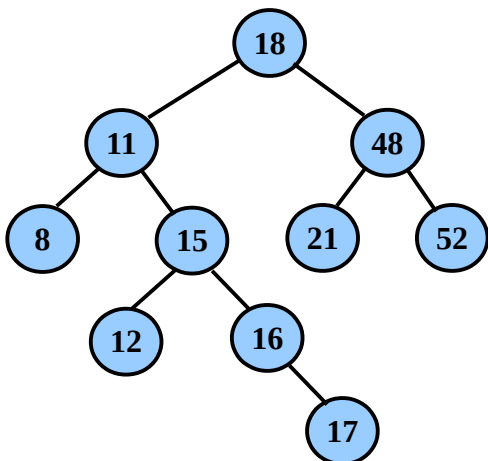Insert: 16 (step 2, right node right rotate)

Insert: 16 (step 3, left rotate)

Insert: 12 (step 1, unbalanced, case 4)

Insert: 12 (step 2, right node right rotate)

Insert: 12 (step 3, left rotate)

Visualization References: https://www.cs.usfca.edu/~galles/visualization/AVLtree.html
http://visualgo.net/bst