MA/CSSE 473 Day 21

AVL Tree
Maximum height

2-3 Trees

Student questions?

Transform and Conquer Algorithms
But first: Answer student questions

- Transform a problem to a simpler instance of the same problem – **instance simplification**
- Transformation to a different representation of the same instance – **representation change**
- Transformation to an instance of a different problem that we know how to solve – **problem reduction**

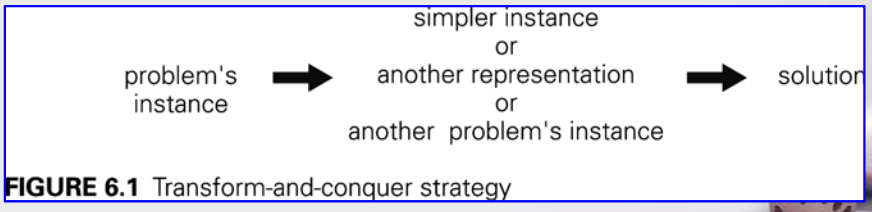

FIGURE 6.1 Transform-and-conquer strategy

Q1

Recap: Representation change: AVL Trees (what you should remember…)

- Named for authors of original paper, **A**delson-**V**elskii and **L**andis (1962).
- An AVL tree is a height-balanced Binary Search Tree.
- A BST T is **height balanced** if T is empty, or if
  - $|\ \text{height}(T_L) - \text{height}(T_R)\ | \leq 1$, and
  - T_L and T_R are both height-balanced.
- Show: Maximum height of an AVL tree with N nodes is $\Theta(\log N)$ **Let's review that together**
- How do we maintain balance after insertion?
- **Exercise:** Given a pointer to the root of an AVL tree with N nodes, find the height of the tree in log N time
- Details on balance codes and various rotations are in the CSSE 230 slides that are linked from the schedule page.

Representation change: 2-3 trees

- Another approach to balanced trees
- Keeps all leaves on the same level
- Some non-leaf nodes have 2 keys and 3 subtrees
- Others are regular binary nodes.

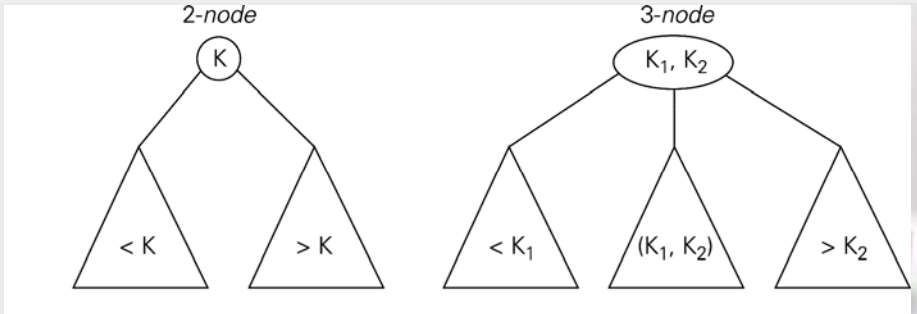

FIGURE 6.7 Two kinds of nodes of a 2-3 tree

2-3 tree insertion example

- More examples of insertion:
  http://www.cs.ucr.edu/cs14/cs14_06win/slides/2-3_trees_covered.pdf
  http://slady.net/java/bt/view.php?w=450&h=300

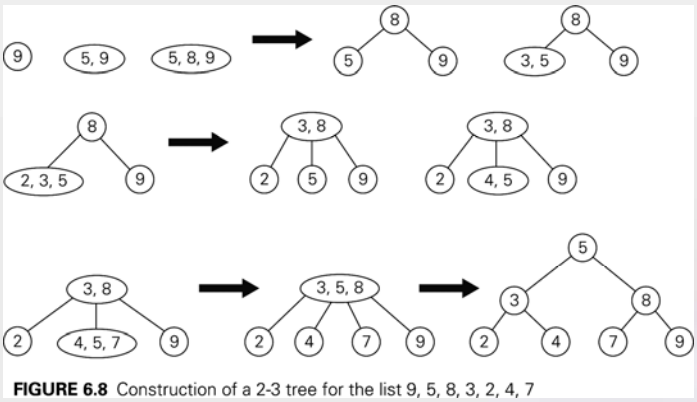

FIGURE 6.8 Construction of a 2-3 tree for the list 9, 5, 8, 3, 2, 4, 7

Add 10, 11, 12, … to the last tree

Efficiency of 2-3 tree insertion

- Upper and lower bounds on height of a tree with n elements?
- Worst case insertion and lookup times is proportional to the height of the tree.

2-3 Tree insertion practice

- Insert 84 into this tree and show the resulting tree