

CSCI 262 Lecture 20 – Binary Search Trees

Outline

- Binary search trees (BSTs) are a type of binary tree (see previous lecture) implementing a *search tree*.
- Search tree – a tree data structure that supports efficient searching for ordered elements.
 - Start at root node – value(s) in root node tell you which child to search.
 - Searching, and typically insertion and deletion as well, have cost proportional to height of tree.
- BST – recursive structure:
 - Nodes contain a single value
 - Left subtree of node contains only values less than node
 - Right subtree contains only values greater than node
- In-order traversal traverses elements in a BST in sort order
- Insertion – always at a leaf, where a search would find the element
- Deletion – three cases, somewhat complex
- Balanced BST – insertions and deletions modified to keep tree balanced so that height is minimized

Readings

Read chapter 10 for Wednesday.

Self Check

1. Binary search trees are the underlying data structure for which higher-level data structures that we've studied?
2. In big O, what is the minimum height of a binary tree in terms of the number of elements stored in it?

For Further Practice

Draw a binary search tree by inserting the following values into it (in the order given): 42, 17, 65, 6, 23, 21, 51, 80, 5, 10.

Do an in-order traversal of the tree.

Finally, try deleting the following values: 80, 23, 17.

If you get stuck or want to check your work, see <https://www.cs.usfca.edu/~galles/visualization/BST.html>