

CS 2100: Data Structures

Homework 8:

Trees, trees, and even more trees!

1. Consider the following two tree traversal outputs:
Postorder: ILOVECOMPUTERS
Inorder: ILVOSCERMOUPET
Draw the binary tree which results in these two outputs for the specified traversals.
2. (Hint: I'd recommend trying this on paper, since you will likely see this on the exams, but feel free to check your answer for correctness, given that you have the code for this data structure!)
 - (a) Draw the AVL tree that results after the following elements are inserted into an initially empty AVL search tree **in this order**: 11, 51, 43, 45, 49, 60, 13, 21, 23, 27
 - (b) Now draw the AVL tree after `remove(49)` is called.
3.
 - (a) Suppose we start with an initially empty max heap, and the following elements are inserted: 25, 11, 17, 33, 30, 21, 7, 15, 23, 42
Draw the exact heap that results after these operations **in this order**. While I don't require you to show your work, I do encourage you to show your work (or at least the intermediate heaps during insertion) for the purposes of partial credit and to check your work.
 - (b) Now draw the max heap that results after `removeMax` is called on your heap from part a. Again, I encourage you to show your work.
 - (c) Finally, draw the *array* representation of your heap from part b.
4.
 - (a) Your classmate claims that the order in which a set of elements is inserted into an AVL tree does not matter - the same search tree results each time. Give a (small) example to show they are wrong.
 - (b) Now this same very wrong classmate claims that a preorder traversal of a heap will list its keys in sorted order. Give a (small) example of a heap that proves he is still wrong.