Algorithms and Data Structures

Summer Semester 2024

For discussion on Wednesday, July 2, 2025

1. (GTG Exercise R-11.2) Insert, into an empty binary search tree, entries with keys 30, 40, 24, 58, 48, 26, 11, 13 (in this order). Draw the tree after each insertion.

2. (GTG Exercise R-11.3) How many different binary search trees can store the keys $\{1, 2, 3\}$?

3. (GTG Exercise R-11.4) Dr. Amongus claims that the order in which a fixed set of entries is inserted into a binary search tree does not matter—the same tree results every time. Give a small example that proves he is wrong.

4. (GTG Exercise R-11.5) Dr. Amongus claims that the order in which a fixed set of entries is inserted into an AVL tree does not matter—the same AVL tree results every time. Give a small example that proves he is wrong.

5. (GTG Exercise R-11.11) Give a schematic figure, in the style of GTG Figure 11.13, showing the heights of subtrees during a deletion operation in an AVL tree that triggers a tri-node restructuring for the case in which the two children of the node denoted as y start with equal heights. What is the net effect of the height of the rebalanced subtree due to the deletion operation?

6. (GTG Exercise R-11.12) Repeat the previous problem, considering the case in which y's children start with different heights.

7. (GTG Exercise R-11.15) What does a splay tree look like if its entries are accessed in increasing order by their keys?

8. (GTG Exercise C-11.29) Explain how to use an AVL tree to sort n comparable elements in $O(n \log n)$ time in the worst case.

9. (GTG Exercise C-11.30) Can we use a splay tree to sort n comparable elements in $O(n \log n)$ time in the worst case? Why or why not?

10. (GTG Exercise C-11.43) Describe a modification to the binary search tree implementation having worst-case $O(1)$-time performance for methods `after(p)` and `before(p)` without adversely affecting the asymptotics of any other methods.