Extra Practice Set 3

True, False, Explain

Decide whether each of the following statements is true or false, and give a reason.

Problem 1. The height of any binary search tree with n nodes is $O(\log n)$.

[True / False]

Solution. False. In the best case, the height of a BST is $O(\log n)$ if it is balanced. In the worst case, however, it can be $O(n)$.

Problem 2. Inserting into an AVL tree with n nodes requires $\log n$ rotations.

[True / False]

Solution. False. Some insertions might not require any rotations to rebalance the AVL tree. Note: $\log n$ instead of $O(\log n)$ is used in the question, therefore, we are referring to exactly $\log n$ rotations instead of at most $\log n$ rotations.

Problem 3. The depths of any two leaves in a max heap differ by at most 1.

[True / False]

Solution. True. A max heap is a complete binary tree.

Problem 4. A tree with n nodes and the property that the heights of the two children of any node differ by at most 2 has $O(\log n)$ height.

[True / False]

Solution. True. This is just a modified AVL tree where the balance factor is between -2 and 2. You can use the same justification for the height of a standard AVL tree to show that this modified AVL tree has height $O(\log n)$.

Problem 5. Given n distinct elements and an empty AVL tree, there is an order in which you can insert the n elements into the AVL tree such that no rotations are required.

[True / False]

Solution. True. You can find an ordering of the elements such that after every insertion, the AVL tree is perfectly balanced. For example, if the elements are the integers 1 to 7, a possible ordering is $4,2,6,1,3,5,7$.

Problem 6. Every directed acyclic graph has exactly one topological ordering.

[True / False]

Solution. False. A graph containing no edges is a valid DAG. This DAG has $n!$ valid topological orderings.

Problem 7. If we double all the edge weights in a directed graph, any shortest path in the original graph will still be a shortest path in the new graph.

[True / False]

Solution. True. You can imagine this doubling as similar to converting from "metres" to "kilometres" - the shortest path between two nodes won't change just because you changed units.

Problem 8. The following array is a max heap: $[10, 3, 5, 1, 4, 2]$.

[True / False]

Solution. False. The element 3 is smaller than its child 4, violating the maxheap property.

Problem 9. In a BST, we can find the next smallest element to a given element in $O(1)$ time.

[True / False]

Solution. False. Finding the next smallest element, the predecessor, may require traveling down the height of the tree, making the running time $O(h)$.

Problem 10. In a weighted undirected tree, depth-first search from a vertex s finds single-source shortest paths from s in $O(V+E)$ time.

[True / False]

Solution. True. In a tree, there is only one path between two vertices, and depth-first search finds it.

Problem 11. Given an array of n numbers in sorted order, an AVL tree on those keys can be built in time $O(n)$.

[True / False]

Solution. True. We can repeatedly find the middle element k, create a node N containing k, and perform recursion on the remaining array elements to create the left and right subtrees of N.

Problem 12. If we square all the edge weights in an undirected graph, any shortest path in the original graph will still be a shortest path in the new graph.

[True / False]

Solution. False. Try to construct a counterexample!

Credits: Problems taken from MIT 6.006 Introduction to Algorithms / 6.046 Design and Analysis of Algorithms