BU CLA CS 530: Analysis of Algorithms

Spring 1996

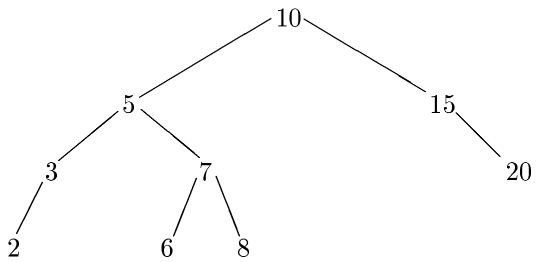
Problem Set 3

Due Thursday 96.02.08, at 11:00 am, in class. Late submissions not accepted.

---

1. Exercise 13.4-2, page 259, in [CLR].

2. A binary search tree is called an AVL tree (after the initials A, V, and L of its inventors) iff the two subtrees of every node differ in height by at most 1. Here is an example of an AVL tree:



   An AVL tree can therefore have a shape quite different from that of a full binary search tree. An AVL tree with n nodes nevertheless has height at most $\mathcal{O}(\log n)$. (AVL trees are not discussed in [CLR].)

   Prove that the height of an AVL tree with $n \geqslant 1$ nodes is at most $1.4404 \log n$. *Hint*: Let $N(h)$ be the minimum number of nodes in an AVL tree of height h. Write a recurrence for $N(h)$, with boundary conditions $N(0) = 1$ and $N(1) = 2$. Show that this recurrence has solution $N(h) > ((1+\sqrt{5})/2)^h$.

3. Exercise 17.2-2, page 336, in [CLR]. *Hint*: Part of the solution was given in lecture.

4. Write the pseudocode for an algorithm that finds the second smallest of $n \geqslant 1$ integers using at most $n + \lceil \log n \rceil - 2$ comparisons, based on the posted solution for the 5th problem in last week problem set. Moreover, *your algorithm must minimize the use of memory storage*, i.e. the space where the input elements are stored. Justify your answer. To simplify the analysis, you can take n to be a power of 2. *Hint*: You can write an iterative algorithm, but a recursive one is perhaps simpler for this problem, at least for the first tournament that finds the smallest of the n elements. Try to adapt algorithm MINIMUM in the first problem set.

5. Exercise 18.2-3, page 363, in [CLR].

Created: 96.01.31