6.856 -- Randomized Algorithms

David Karger

Handout #3, September 11, 2000 -- Homework 2, Due 9/18

M.R. refers to this text:
Motwani, Rajeez, and Prabhakar Raghavan. *Randomized Algorithms*. Cambridge: Cambridge University Press, 1995.

1. MR 1.8.

2. MR 2.3. Consider a uniform rooted tree of height h (every leaf is at distance h from the root). The root, as well as any internal node, has 3 children. Each leaf has a boolean value associated with it. Each internal node returns the value returned by the majority of its children. The evaluation problem consists of determining the value of the root; at each step, an algorithm can choose one leaf whose value it wishes to read.
    (a) Show that for any deterministic algorithm, there is an instance (a set of boolean values for the leaves) that forces it to read all $n = 3^h$ leaves.

    (b) Show that there is a nondeterministic algorithm can determine the value of the tree by reading at most leaves. In other words, prove that one can present a set of this many leaves from which the tree value can be determined.

    (c) Consider the recursive randomized algorithm that evaluates two subtrees of the root chosen at random. If the values returned disagree, it proceeds to evaluate the third sub-tree. Show the expected number of leaves read by the algorithm on any instance is at most $n^{0.9}$.

3. MR 2.6. Use Yao's minimax principle to prove a lower bound on the expected running time of any Las Vegas algorithm for sorting n numbers that uses only comparisons. You might want to review deterministic sorting lower bounds from, e.g., [CLR] chapter 9.

4. *(optional)* MR 1.15. Prove that $NP \subseteq BPP$ implies $NP = RP$.

Bibliography

CLR

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, *Introduction to Algorithms*.