Homework #2: Recurrences, Sorting, Selection, Lower Bounds
Due Date: Tuesday, 9 July 1996

Reading

Read Ch. 3, 4, 9.1–9.3 and 10.1 of CLR.

Practice

Recall, these exercises are purely for your own practice. You need only turn in the official problems.

- Do CLR 3.1-5.
- Do CLR 4.2-4, 4.2-5.
- Do CLR 9.1-3.
- Do CLR 9.1-5.
- Do CLR 9.3-4.

Problems

1. (20 points) Do CLR 4-1.
2. (20 points) Do CLR 4-2. To make the analysis cleaner, we will charge your algorithm only for the number of "bit-fetch" operations. The input to the algorithm consists of an array of n numbers, each of which is comprised of $\lceil \lg n \rceil$ bits. Give an algorithm which can find the missing number by fetching only $O(n)$ bits.
3. (40 points) Do CLR 9-1 (parts a–e only).
4. (20 points) Do CLR 9.3-3.

Extra Credit

In problem 10.1-1 (discussed in lecture), we gave an algorithm which finds the second smallest of n elements using $n + \lceil \lg n \rceil - 2$ comparisons in the worst case. Here, we want you to show that this algorithm is almost optimal by proving a general lower bound showing that $n + \lfloor \lg n \rfloor - 2$ comparisons are required in the worst case for *any* algorithm which finds the second smallest element.