

Due Thursday, April 18, 2013 (but you can turn it in by Monday morning April 22).

Problem 1. Assume you use the selection algorithm from class (and from CLRS) but we use columns of size 13. Assume we use *selection sort* to find the median of each column.

- (a) Briefly list each step of the algorithm and how many comparisons the step takes.
- (b) Write a recurrence for the number of comparisons the algorithm uses.
- (c) Solve the recurrence using constructive induction. Just get the high order term exactly.
- (d) How does this value compare to what we got in class with columns of size 5?

Problem 2. Assume you have an algorithm that finds the median of n elements in $cn \lg \lg n - n$ comparison steps (for some constant c).

- (a) Give an efficient (recursive) algorithm for selection based on this. It will, of course, not be linear time.
- (b) Write a recurrence for the number of comparisons your algorithm uses.
- (c) Solve the recurrence using constructive induction. Just get the high order term exactly.
- (d) Why might it be a good algorithm despite not being linear time?

Problem 3. Make an intuitive argument for why no algorithm for selection can use fewer than $2n$ comparisons in the worst case. [If you feel that $2n$ is not a good choice, but think that you can make an intelligent argument for some other reasonable value, such as $1.5n$ or $2.5n$, then do so. (An argument for $1n$ is not reasonable.)]