Fall 2012 **CMSC 351: Homework 6** Clyde Kruskal

Due at the start of class Friday, October 19, 2012.

Problem 1. Consider an array of size eight with the numbers $80, 30, 40, 70, 10, 20, 60, 50$. Assume you execute quicksort using the version of partition from CLRS. Note that in this algorithm an element might exchange with itself (which counts as one exchange).

(a) Show the array after the first partition. How many comparisons and exchanges are used?

(b) Show the left side after the next partition. How many comparisons are used? How many exchanges?

(c) Show the right side after the next partition on that side. How many comparisons are used? How many exchanges?

Problem 2. Consider the following quicksort-like sorting algorithm. Pick two elements of the list. Partition based on both of the elements. So the elements smaller than both are to the left, the elements in between are in the middle, and the elements larger than both are to the right.

(a) Given a brief English description of how you would partition. Write high level pseudo code for this algorithm. Try to minimize the number of comparisons.

(b) How many comparisons does the partition algorithm use in the worst case? Justify informally.

(c) How many comparisons does the partition algorithm use on average. Justify informally.

(d) How many exchanges (or moves) does the partition algorithm use in the worst case? Justify informally.

(e) Assume that the two partition elements always partition exactly into thirds. Write a recurrence for the number of comparisons. Solve this recurrence using constructive induction. Just get the high order term exactly.

(f) Assume that the two partition elements always partition so exactly one quarter are to the left, one half in the middle, and one quarter to the right. Write a recurrence for the number of comparisons. Solve this recurrence using constructive induction. Just get the high order term exactly.

(g) CHALLENGE PROBLEM: Find the exact high order term for the average number of comparisons.