Due at the start of class Wednesday, June 26, 2002.

Problem 1. Consider an array of size eight with the numbers $50, 70, 10, 20, 60, 40, 80, 30$. Assume you execute quicksort using the version of partition from CLRS.

(a) What is the array after the first partition. How many comparisons did you use? How many exchanges?

(b) Show the left side after the next partition. How many comparisons did you use? How many exchanges?

(c) Show the right side after the next partition on that side. How many comparisons did you use? How many exchanges?

(d) What is the total number of comparisons in the entire algorithm? What is the total number of exchanges in the entire algorithm?

Problem 2. Assume you execute quicksort using the version of partition from CLRS.

(a) What is the most exchanges that the algorithm will execute for an input of size n. Give an example with $n = 8$.

(b) What is the fewest exchanges that the algorithm will execute for an input of size n. Give an example with $n = 8$.

Problem 3. 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 inbetween are in the middle, and the elements larger than both are to the right.

a. Write high level code for this algorithm.

b. How many comarisons and exchanges does the partition algorithm use?

c. Assume the partition elements always partition exactly into thirds. Write a recurrence for the number of comparisons. "Solve" this recurrence.

d. Assume the 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.

e. CHALLENGE PROBLEM. Can you refine your analysis of this version of quicksort? For example, can you find its average case?

Problem 4.

(a) Illustrate the operation of radix sort on the following list of English words: RUT, TOP, SUN, POT, TON, OPT, TOR, ROT, TOO, OUT, SUP, PUT

(b) Write an English sentence using both "tor" and "rot" (that indicates you understand the meanings of both words).

Problem 5. Challenge problem. Consider the following recurrence that basically occurs from one of the recursive integer multiplication algorithms:

$$T(n) = \begin{cases} 4T(\lceil n/2 \rceil) + n & \text{if } n > 1 \\ 0 & \text{if } n = 1 \end{cases}$$

Find at constant a such that $T(n) = an^2 + O(n)$. Justify your answer.