Problem Set 2—Due Monday, April 23 3:15 PM

(20) Problem 1.

parts a and g of Problem 4–1 (pp. 107).

(15) Problem 2. Solve the following recurrence relation to big Theta accuracy:

$$\begin{aligned} T(n) &= 1 \quad n = 1 \\ &= T(n/2) + T(n/5) + 2n \quad n > 1 \end{aligned}$$

(25) Problem 3. Problem 4-5 page 109.

(20) Problem 4. Problem 5.2-5 on page 122. Expected number of inversions.

(20) Problem 5.

In our discussion of the Select algorithm we assumed there were no duplicate items in our input list. We now consider the more general case where our input list of numbers, $x_1, x_2, \cdots x_n$ may have multiple copies of the same item.Note that even with duplicates, our notion of the kth smallest element remains the same: the value of the kth element of a sorted list: e.g. for $1, 3, 3, 3, 3, 4, 4, 8, 8, 9$, return 3 if $k = 2, 3, 4$, or 5, and return 4 if $k = 6, 7$.

(a) The algorithm described in class (and in the text) will work correctly even if there are many duplicates, however it may be much slower. Analyze the running time of our algorithm to find the median element when the list consists of n copies of the same item. Obviously which element you choose as the split element doesn't matter in this case, so the best case, average case, and worst case are all the same. Note that with duplicates the "SMALL" set is assigned all other elements equal to or less than the split element (so after partitioning we have SMALL, then the split element, then BIG which is those strictly greater).

(b) Modify the randomized-select algorithm so it has good expected performance even when there may be many duplicates (HINT: modify the partition routine along with the method of deciding which sublist to continue searching).