1. In class we showed using an adversary argument that any algorithm to compute the MAX and the MIN of a set of $n$ distinct numbers simultaneously using pairwise comparisons must, in the worst-case, use at least $\lceil 3n/2 \rceil - 2$ comparisons.

   (a) Use the decision tree argument (finding a lower bound on the number of possible responses, which is a lower bound on the number of leaves, and using this as a bound on the height of the tree) to develop a worst-case lower bound on the number of pairwise comparisons.

   (b) If this bound is different than $\lceil 3n/2 \rceil - 2$, explain how seemingly contradictory bounds can both be correct.

(20 points)

2. Show that the second largest element can be found with $n + \lceil \log n \rceil - 2$ comparisons in the worst case. (20 points)

3. Show how QUICKSORT can be made to run in $O(n \log n)$ time in the worst case, assuming that all elements are distinct. (20 points)

4. Show that $2n - 1$ comparisons are necessary in the worst case to merge two sorted lists containing $n$ elements each. (20 points)

5. In our linear-time selection algorithm, the inputs are divided into groups of 5. What if you used groups of 3 instead? What if used groups of 7 or larger (odd integers) (Explain)? (20 points)