Algorithms — Homework 3
Sorting

Due: September 22.

1) Sort the following sequence using merge sort and quicksort.

$$22 \quad 80 \quad 18 \quad 9 \quad 90 \quad 12 \quad 22 \quad 57 \quad 86 \quad 36 \quad 32 \quad 88 \quad 20 \quad 6 \quad 62 \quad 22$$

   For merge sort, show subsequences before and after merging. For quicksort, show the selected pivot element and the resulting partition.

2) For each of the following problems, give an algorithm that finds the desired numbers within the given amount of time.
   (a) Let S be an *unsorted* array of n integers. Give an algorithm that finds the pair $x, y \in S$ that *maximizes* $|x - y|$. Your algorithm must run in $\mathcal{O}(n)$ worst-case time.
   (b) Let S be a *sorted* array of n integers. Give an algorithm that finds the pair $x, y \in S$ that *maximizes* $|x - y|$. Your algorithm must run in $\mathcal{O}(1)$ worst-case time.
   (c) Let S be an *unsorted* array of n integers. Give an algorithm that finds the pair $x, y \in S$ that *minimizes* $|x - y|$, for $x \neq y$. Your algorithm must run in $\mathcal{O}(n \log n)$ worst-case time.
   (d) Let S be a *sorted* array of n integers. Give an algorithm that finds the pair $x, y \in S$ that *minimizes* $|x - y|$, for $x \neq y$. Your algorithm must run in $\mathcal{O}(n)$ worst-case time.

3) Give an efficient algorithm to compute the union of sets A and B, where $n = \max(|A|, |B|)$. The output should be an array of distinct elements that form the union of the sets, such that they appear exactly once in the union. Assume that A and B are unsorted. Give an $\mathcal{O}(n \log n)$ time algorithm for the problem.