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

22  80  18  9  90  12  22  57  86  36  32  88  20  6  62  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 $O(n)$ 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 $O(1)$ 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 $O(n \log n)$ 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 $O(n)$ 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 $O(n \log n)$ time algorithm for the problem.