CSCI 533 Analysis of Algorithms
Homework #3
Due Thursday, September 21st at Midnight

1. An $n$-element array contains only the numbers 0, 1 and 2. Write an $O(n)$ algorithm to sort the numbers. Legal operations on the data are swapping two elements in the array and testing whether an element in the array is 0, 1, 2. So, something like counting sort will not work.

2. The $k$th quantiles of an $n$-element set are the $k-1$ order statistics that divide the sorted set into $k$ equal-sized sets (to within 1). Give an $O(n \log k)$-time algorithm to list the $k$th quantiles of a set. [CLRS 9.3-6, p. 192]

3. Let $X[1..n]$ and $Y[1..n]$ be two arrays, each containing $n$ numbers already in sorted order. Give an $O(\log n)$-time algorithm to find the median of all $2n$ elements in arrays $X$ and $Y$. [CLRS 9.3-8, p. 193].

4. Red-black trees:
   (a) Show the red-black trees that result after successively inserting the keys 41, 38, 31, 12, 19, 8 into an initially empty red-black tree. [CLRS 13.3-2, p. 287]

   (b) Consider a red-black tree formed by inserting $n$ nodes with RB-INSERT. Argue that if $n > 1$, the tree has at least one red node. [CLRS 13.3-5, p 287]

   (c) In Exercise 13.3-2, you found the red-black tree that results from successively inserting the keys 41, 38, 31, 12, 19, 8 into an initially empty tree. Now show the red-black trees that result from the successive deletion of the keys in the order 8, 12, 19, 31, 38, 41.

5. The nuts and bolts problem is defined as follows. You are given a collection of $n$ bolts of different widths, and $n$ corresponding nuts. You are allowed to try a nut and bolt together, from which you can determine whether the nut is too large, too small, or an exact match for the bolt, but there is no way to compare two nuts together, or two bolts together. You are to match each bolt to its nut. Devise an algorithm for the nuts and bolts problem that runs in time $O(n \log n)$ on average.