For full credit, please adhere to the following:

- Unsupported answers receive no credit.

- All answers can be typed or handwritten, and should be readable.

- Submit the assignment in one pdf file

1. (20 points) **Exercise 5.1**: You are interested in analyzing some hard-to-obtain data from two separate databases. Each database contains $n$ numerical values – so there are $2n$ values total – and you may assume that no two values are the same. You'd like to determine the median of this set of $2n$ values, which we define here to be the $n^{th}$ smallest value.

   However, the only way you can access these values is through *queries* to the databases. In a single query, you can specify a value $k$ to one of the two databases, and the chosen database will return the $k^{th}$ smallest value that it contains. Since queries are expensive, you would like to compute the median using as few queries as possible.

   Give an algorithm that finds the median value using at most $O(\log n)$ queries.

2. (20 points) **Exercise 5.2**: Recall the problem of finding the number of inversions. As in the text, we are given a sequence of $n$ numbers $a_1, \ldots, a_n$ which we assume are all distinct, and we define an inversion to be a pair $i < j$ such that $a_i > a_j$.

   We motivated the problem of counting the number of inversions as a good measure of how different two orderings are. However, one might feel this measure is too sensitive. Let's call a pair a *significant inversion* if $i < j$ and $a_i > 2a_j$. Give an $O(n\log n)$ algorithm to count the number of significant inversions between two orderings.

3. (20 points) **Exercise 5.3**: Suppose you're consulting for a bank that's concerned about fraud detection, and they come to you with the following problem. They have a collection of $n$ bank cards that they've confiscated, suspecting them of being used in fraud. Each bank card is a small plastic object, containing a magnetic stripe with some encrypted data, and it corresponds to a unique account in the bank. Each account can have many bank cards corresponding to it, and we'll say that two bank cards are *eqivalent* if they correspond to the same account.

   It's very difficult to read the account number off a bank card directly, but the bank has a high-tech "equivalence tester" that takes two blank cards and, after performing some computations, determines whether they are equivalent.

Their question is the following: among the collection of $n$ cards, is there a set of more than $n/2$ of them that are all equivalent to one another? Assume that the only feasible operations you can do with the cards are to pick two of them and plug them in to the equivalence tester. Show how to decide the answer to their question with only $O(n\log n)$ invocations of the equivalence tester.

4. (20 points) **Exercise 5.6**: Consider an $n$-node complete binary tree, $T$, where $n = 2^d - 1$ for some $d$. Each node $v$ of $T$ is labeled with a real number $x_v$. You may assume that the real numbers labeling the nodes are all distinct. A node $v$ of $T$ is a *local minimum* if the label $x_v$ is less than the label $x_w$ for all nodes $w$ that are joined to $v$ by an edge.

   You are given such a complete binary $T$ but the labeling is only specified in the following *implicit* way: for each node $v$, you can determine the value $x_v$ by *probing* the node $v$. Show how to find a local minimum of $T$ using only $O(\log n)$ *probes* to the nodes of $T$.

5. (20 points) **Exercise 5.7**: Suppose now that you're given an $n \times n$ grid graph $G$. (An $n \times n$ grid graph is just the adjacency graph of an $n \times n$ chessboard. To be completely precise, it is a graph whose node set is the set of all ordered pairs of natural numbers $(i, j)$, where $1 \leqslant i \leqslant n$ and $1 \leqslant j \leqslant n$; the nodes $(i, j)$ and $(k, l)$ are joined by an edge if and only if $|i - k| + |j - l| = 1$.)

   We use some of the terminology of the previous question. Again, each node $v$ is labeled by a real number $x_v$; you may assume that all of these nodes are distinct. Show how to find a local minimum of $G$ using only $O(n)$ probes to the nodes of $G$. (Note that $G$ has $n^2$ nodes.)