EE 360C — Algorithms — Summer 2013
Homework #4

Due: July 17, 2013 11:30am (in class)

Homework problems are to be done individually. You may discuss the problem and general concepts with other students, but you must write your solutions independently.

Each question is worth 10 points. Maximum possible score is 30.

Whenever you give an algorithm, prove that it is correct.

---

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 will 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. Suppose you're consulting for a bank thats 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 *equivalent* 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 bank 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.

3. Give an expression for $T(n)$ if the given recurrence can be solved with Master Theorem. Otherwise indicate that hte Theorem does not apply.

   (a) $T(n) = 3T(n/2) + n^2$

   (b) $T(n) = 4T(n/2) + n^2$

   (c) $T(n) = 16T(n/4) + n$

   (d) $T(n) = 2T(n/4) + n^{0.51}$

   (e) $T(n) = 2nT(n/2) + n$

   (f) $T(n) = 2T(n-2) + n$