CS466/666, Fall 2011: Assignment 2

Out: October 5, Due: October 19, 5pm

1. **Finding minimum and maximum:** (10 marks) Recall from class that finding the minimum and maximum of a given set of n items takes at least $\frac{3}{2}n - 2$ comparisons (for any deterministic, comparison-based algorithm.)

   Design a (deterministic, comparison-based) algorithm that finds the minimum and maximum of a given set of n items and uses only $\lceil \frac{3}{2}n \rceil - 2$ comparisons between the items.

2. **Independent set:** (14 marks) An *independent set* of a graph G is a set of vertices I such that no two vertices are adjacent. Consider the following algorithm for finding an independent set:

   1. Compute $d = 2m/n$.
   2. For each vertex $v \in V$, add v to an (initially empty) set I_1 with probability $1-1/d$.
   3. Delete all vertices in I_1 (and their incident edges.)
   4. For each edge e that remains in the graph, add one (arbitrary) endpoint of e to an (initially empty) set I_2.
   5. Return $I := V - I_1 - I_2$.

   This assignment guides you to show that this yields a good independent set. Some of the steps below are nearly trivial, but write a sentence or two anyway.

   (a) Argue that I is an independent set.

   (b) Argue that the expected size of I_1 is $n - n/d$.

   (c) Argue that at step 4, the expected number of edges left in the graph is $n/2d$.

   (d) Argue that the expected size of I_2 is no more than $n/2d$.

   (e) Conclude that the expected size of the independent set is at least $n^2/4m$.

3. **Verification of matrix inverse** (10 marks) Let $A = (a_{i,j})$ be an $n \times n$-matrix. Recall that the *inverse matrix* of A is the $n \times n$-matrix A^{-1} such that $A \cdot A^{-1}$ is the identity matrix.

   Presume you are given two $n \times n$-matrices A and B. Give an $O(n^2)$ Monte-Carlo algorithm to test whether $B = A^{-1}$. Your algorithm should have error-probability at most $\frac{1}{2}$.

4. **Intersection of half-planes** (6 marks) Let $H = \{h_1, \ldots, h_n\}$ be a set of n half-planes ~~where $h_i = \{(x,y) : y \geq m_i x + b_i\}$.~~ where $h_i = \{(x,y) : y \geq m_i x + b_i\}$ for $i = 1, \ldots, k$, and $h_i = \{(x,y) : y \leq m_i x + b_i\}$ for $i = k+1, \ldots, n$, In other words, you are given the $2n$ numbers $m_1, \ldots, m_n$ and $b_1, \ldots, b_n$ as well as the index k.

   Design a randomized Las Vegas algorithm that tests in $O(n)$ expected time whether there exists any point (x^*, y^*) that belongs to all half-planes.