COS 521: Advanced Algorithms
Final Exam Spring 2011
Moses Charikar
Due May 18 at the latest.

Instructions: The final has 9 problems. Solve any 7 of your choice.

Finish the test within **48 hours** after first reading it. You can consult any notes/handouts from this class and feel froo to quote, without proof, any results from there. You should not consult any other source or person in any way.

Do not read the test before you are ready to work on it.

Write and sign the honor code pledge on your exam (The pledge is "I pledge my honor that I have not violated the honor code during this exam and followed all instructions.")

1. Consider the problem of scheduling n jobs on a single machine where each job j has a processing time p_j and a weight $w_j > 0$, and the goal is to minimize the weighted sum of completion times $\sum_{j=1}^{n} w_j C_j$. The machine can execute only one job at a time and a job once started, must be processed to completion. Show that it is optimal to schedule jobs in decreasing order of w_j/p_j.

2. A slight variant of the k-median problem discussed in class is this: Given a set of points X, parameter k and distances c_{ij} between points in X., the goal is to select k centers in X such that the sum of distances of each point to its nearest center is minimized, i.e. find $S \subseteq X$, $|S| = k$ so as to minimize $\sum_{j \in X} \min_{i \in S} c_{ij}$.

   Give a polynomial time algorithm for this problem when the set of points X are vertices of a tree T and the distances c_{ij} are shortest path distances on T. (Hint: Use dynamic programming. It may be helpful to assume that T is a rooted binary tree, but show that this is without loss of generality.)

3. Consider the *facility location with penalties* problem where we are given a set of clients D, a set of facilities F, distance c_{ij} between client j and facility i, and cost f_i for opening a facility at i. In addition to this, every client j is associated with a penalty p_j and for every client j, the algorithm has the choice of either
   (1) connecting it to some facility i (incurring cost c_{ij}), or
   (2) not connecting it to any facility and paying penalty p_j.
   Thus the total cost of the algorithm is the sum of three terms:
   1. facility costs for facilities opened,
   2. connection costs for clients connected to facilities, and
   3. penalties for clients not connected.

   (a) Write down an LP relaxation for this problem. Also write down the dual LP.

   (b) Adapt the primal-dual approximation for facility location to get a 3 approximation for *facility location with penalties*. You only need to indicate briefly how the analysis changes.

4. Given a 3-colorable graph $G(V, E)$ on n vertices, give a polynomial time algorithm to pick a subset $S \subseteq V$ of size $n/2$ such that the induced subgraph on S does not contain a clique of size 3. (Hint: Consider the SDP for graph coloring and think about SDP rounding algorithms we have seen in class.)

5. This is a variation of the Weighted-Majority algorithm for the experts problem:

   (a) Each expert begins with weight 1 (as before).

   (b) We predict the result of a weighted majority vote of the experts (as before).

   (c) If an expert makes a mistake, we penalize him by dividing the weight by 2, but only if the weight was at least 1/4 of the average weight of experts (this last twist is new).

   Prove that in any contiguous block of trials, the number of mistakes made by the algorithm is at most $O(m + \log n)$ where m is the number of mistakes made by the best expert in that block and n is the total number of experts.

6. In the Steiner tree problem, given a graph $G(V, E)$ and a set of terminals $X \subseteq V$, the goal is to find a minimum cost tree that connects the terminals, possibly including vertices outside X. In the online version of this problem, we know the underlying graph at the outset, but terminals are revealed in an online fashion. The goal is to maintain a low cost Steiner tree that connects all terminals revealed so far. Consider the following greedy algorithm for online Steiner tree: When a new terminal v arrives, we connect it using the shortest path to a vertex in the current tree. We will analyze the cost of the tree produced by the algorithm after n terminals arrive.

   (a) Let C_k be the minimum distance of the kth terminal to the first $k-1$ terminals. (Note that $C_1 = 0$.) Show that the cost incurred by the algorithm to connect the kth terminal to the tree is at most C_k.

   (b) Let OPT be the cost of the optimal Steiner tree on the n terminals. Let $\hat{C}_i$ be the ith largest cost in the set $\{C_1, C_2, \ldots, C_n\}$. Show that $\hat{C}_k \leq 2OPT/k$. (Hint: Consider the terminals associated with the k largest costs. What can you say about their pairwise distances ?).

   (c) Prove an upper bound on the competitive ratio of the greedy algorithm.

7. The minimum r-way cut problem is the problem of partitioning a graph into r pieces by removing the smallest set of edges. Note that the minimum 2-way cut is just the min cut problem. Consider adapting Karger's contraction algorithm for min cut to solve the minimum r-way cut problem: We apply the contraction procedure iteratively as before, stop when the contracted graph has exactly r vertices, and consider the corresponding r-way partition of the vertices of the original graph. Show that the probability that this procedure produces a minimum r-way cut is $\Omega(n^{-2(r-1)})$.

8. In order to implement a counter capable of counting from 1 to N, at least $\log N$ bits are needed. In this question you'll analyze a "probabilistic counter" - a counter that counts with small errors, and uses only $\Theta(\log \log N)$ bits.

   The counter is defined as follows:

   - Initially the counter is 0.
   - An INCREMENT operation is performed as follows: If the current value of the counter is i, increase the counter value by 1 with probability $\frac{1}{2^i}$, and leave the counter in its current state (i) with probability $1 - \frac{1}{2^i}$.

   Let C_k be a random variable denoting the value of the counter after k INCREMENT operations have occurred.

   (a) Suppose that k INCREMENT operations have been performed. Prove that

   $$E[2^{C_k}] = k + 1.$$

   (Hint: Define the random variable $X_i := 2^{C_i} - 2^{C_{i-1}}$ and compute $E[X_i | C_{i-1} = t]$.)

   (b) Compute the variance of 2^{C_k}.
   (Hint: Define $Y_i := 4^{C_i} - 4^{C_{i-1}}$ and compute $E[Y_i | C_{i-1} = t]$.)

   (c) Show that using a small number of independent such counters, after k increment operations, one can obtain an estimate within $k(1 \pm \varepsilon)$ with probability $1 - \delta$. Give a bound on the number of counters needed as a function of ε and δ.

9. In this problem, you will analyze *beacon-based distance estimation* which is a technique used to estimate distances (i.e. latencies) on the internet using a small number of distance measurements. For every pair of nodes u, v, we have a distance d_{uv} that we are trying to estimate. Assume that these distances satisfy triangle inequality. The set of beacons B is a subset of the set of nodes. Suppose that for each node u and beacon $b \in B$, we know the distance d_{ub}. From this data, we obtain an estimate on the distance between u and v: $\hat{d}_{uv} = \min_{b \in B} d_{ub} + d_{vb}$.

   Show that, if you pick a suitably large subset of beacons at random, then with probability $1 - \gamma$, for at least $1 - \epsilon$ fraction of node pairs (u, v), $\hat{d}_{uv} \leq 3 d_{uv}$. Find an expression for the number of beacons you need as a function of ϵ and γ. (Hint: Consider a ball around u containing $\epsilon/2$ fraction of the nodes.)