# Algorithms, Final, Jan 3 2017 (3.5 hours, total 40pts)

January 2, 2017

You can use any NPC problem we mentioned in class or homeworks for your reductions.

$*$ Please answer eight questions from the following ten questions. If you answer more than eight questions, we will give you scores according to the eight problems for which you get the highest scores.

You can use the following version of Chernoff Bound. (Chernoff Bound) Let $X_1, \dots, X_n$ be $n$ i.i.d. 0/1 random variables. Let $\mu = \mathbb{E}[X_i]$. Then for any $\epsilon > 0$, we have that

$$\Pr\left[\left|\frac{1}{n}\sum_{i=1}^{n} X_i - \mu\right| \geq \epsilon\right] \leq 2\exp\left(-\frac{n\epsilon^2}{2}\right).$$

Other versions that I used in the class should work equally well for this exam. You can use any of them.

1. (5pts) The post-office location problem is defined as follows. We are given $n$ points $p_1, \dots, p_n$ with associated weights $w_1, \dots, w_n$. We wish to find a point $p$ (not necessarily one of the input points) that minimizes the sum $\sum_{i=1}^{n} w_i d(p, p_i)$, where $d(a, b)$ is the distance between points $a$ and $b$.

   Try to solve the 1-dimensional post-office location problem in linear time and prove the correctness. (If your algorithm is randomized, the expected running time should be linear.)

2. (5 pts) (Exact Cover) We are given a set $U$ of elements and a family $\mathcal{F}$ of subsets of $U$. The *exact cover* problem asks whether there is a subset $S$ of $\mathcal{F}$ such that all subsets in $S$ are disjoint and their union is $U$.

   (a) Show this problem is NPC. (2.5pts)

   (b) If every subset in $\mathcal{F}$ contains only two elements in $U$, show the problem is polynomial solvable. (2.5pts)

3. (5 pts) There are $k$ vacation periods (e.g., national days, spring festival), each spanning several contiguous days. Let $D_j$ be the set of days included in the $j$th vacation period. We need to assign $n$ doctors at the hospital to cover a set of vacation days. Suppose doctor $i$ has a set of $S_i$ of vacation days when she/he is available to work. Moreover, each doctor should be assigned to work at most $c$ vacation days in total ($c$ is a given number). For each vacation period $j$, each doctor should work at most for one day in $D_j$. Design a polynomial time algorithm to find such an assignment (if exists).

4. (5pts) There are $m$ machines and $n$ jobs. Jobs have identical processing needs. The machines are different. Machine $i$ has a parameter $\ell_i$, which is the time it needs to process a single job. If we finish

job $j$ at time $t$, we pay a cost of $c_j(t)$ ($c_j(t)$ is nonnegative and increasing). Each machine can process one job at a time. Design a polynomial time algorithm that finds a min-cost schedule (a schedule specifies which jobs are assigned to which machine and their order).

5. (5pts) You are given a coin some bias. Let $p$ be the probability that a toss of the coin results in a head. Let $\Delta = |p - 1/2|$. $\Delta \neq 0$. Your goal is to decide whether $p > 1/2$ or not.

   (1) (1.5pts) Suppose you know $\Delta$ (but not $p$). Show that $O(\Delta^{-2})$ tosses is enough to do the job with probability 0.99.

   (2) (3.5pts) What if you don't know $\Delta$? Provide an algorithm which uses $O(\Delta^{-2} \log \log \Delta^{-1})$ tosses (it should succeed with probability 0.99). (hint: Guess $\Delta$. Try to allocate the success probability wisely for each guess and use union bound. You can get partial credits if you get worse bound.)

6. (5pts) There is a set of pages $P = \{1, 2, ..., n\}$ that can be broadcast by a server. Assume that time is discrete and there are $T$ time slots. At each time slot, the broadcast server can broadcast exactly *one* page and all of the users could receive that page. There are several users. Each user $u$ is associated with a page $p_u \in P$ and a time interval $[b_u, e_u]$. $u$ can be satisfied if the server broadcasts page $p_u$ any time during $[b_u, e_u]$.

   Design an algorithm to schedule the broadcast server in order to satisfy as many users as possible. The problem is NP-hard (you do not have to prove it). Design an approximation algorithm that can achieve a constant approximation. (hint: use LP rounding).

7. (5pts) You start with a set $X$ of $n$ actresses and a set $Y$ of $n$ actors, and two players P0 and P1. Player P0 names an actress $x1_{\in}X$, player P1 names an actor $y_1$ who has appeared in a movie with $x_1$, player P0 names an actress $x_2$ who has appeared in a movie with $y_1$, and so on. Thus, P0 and P1 collectively generate a sequence $x_1, y_1, x_2, y_2, \ldots$ such that each actor/actress in the sequence has costarred with the actress/actor immediately preceding. A player Pi(i = 0, 1) loses when it is Pi?s turn to move, and she cannot name a member of her set who hasn?t been named before.

   Suppose you are given a specific pair of such sets $X$ and $Y$ , with complete information on who has appeared in a movie with whom. A strategy for Pi in our setting is an algorithm that takes a current sequence $x_1, y_1, x_2, y_2, \ldots$ and generates a legal next move for Pi (assuming it?s Pi?s turn to move). Give a polynomial time algorithm that, given some instance of the game, decides at the start of the the game which of the two players can force a win. Provide detailed explanation why your algorithm is correct. (Big Hint: First think about what happens when there is a perfect matching?)

8. We are given an undirected graph $S$, where each node is of degree $d$. Recall that a dominating set is a subset of vertices $S$ such that for each vertex $u \notin S$, there is an edge $(u, v)$ with $v \in S$. It is easy to see that a dominating set of $S$ should contain at least $\frac{n}{d+1}$ nodes. Show that a set of $O(n \log n/d)$ nodes chosen uniformly at random from $G$ is a dominating set with high probability (Note this implies an extremely simple $O(\log n)$ approximation). Provide the detail of your analysis.

9. (5pts) In class, we learnt that finding $k$ edge/vertex-disjoint paths from $s$ to $t$ can be solved using flow. However, the following variant turns out to be NPC. Given a directed graph $G$ and $k$ pairs of nodes $(s_1, t_1), \ldots, (s_k, t_k)$. The problem asks whether there exists $k$ vertex-disjoint path $P_1, \ldots, P_k$ such that $P_i$ goes from $s_i$ to $t_i$. Prove the problem is NPC.

10. (5pts) (1) (2pts) We are given a bipartite graph. We would like to find a maximum matching of cardinality exactly $k$. Show the following polytope is integral (hence we can do it by solving the LP):

$$\max . \sum_e w_e x_e \quad \sum_e x_e = k; \quad \sum_{e \in \delta(v)} x_e \leq 1 \;\; \forall v; \quad x_e \in [0,1].$$

Here $\delta(v)$ is the set of edges incident on vertex $v$.

(2) (3pts) We are given a bipartite graph where each node has degree exact $d$. Show that there is a perfect matching, using the polytope theory we taught in the class (we prove in the class that we can do this using Hall's theorem).