CSCI 270 Homework #5

Due Date: Tuesday, November 19th, 2pm

Submit in class or the dropbox (Box 3, first floor of SAL, opposite the Men's bathrooms).

1. Write your name, student ID Number, and which lecture you attend (morning or afternoon). Multi-page submissions must be stapled.

2. Special Agent Ethan Hunt wants to infiltrate the headquarters of Evil Mastermind Owen Davian. Owen's headquarters are mapped out via an undirected graph G, where the nodes are intersections and the edges are corridors. The entrance to the headquarters is the node s, and Owen is located at node t. Owen wants to place some deadly booby traps in various corridors of his headquarters in such a way that Ethan must go past at least one of these booby traps on the way from s to t.

   Give a polynomial-time algorithm which determines the minimum number of booby-traps needed, and which corridors to place them in. Justify the correctness of your algorithm.

3. In a room, there are n light fixtures $\{l_1, ..., l_n\} = L$ and n outlets $\{o_1, ..., o_n\} = O$. Each fixture l_i is in view of some subset of the outlets $S_i \subseteq O$. You want to determine if the room is *ergonomic*, that is, you can plug each light into a unique outlet such that if l_i is plugged into o_j, then $o_j \in S_i$. Give a polynomial-time algorithm to determine if the room is *ergonomic*.

4. There are n courses in the Computer Science program, and in order to graduate, a student must satisfy several requirements. Each requirement is of the form"you must take at least k_i courses from the subset S_i". The difficult part is that you cannot count a single course for multiple requirements. For example, if you need to take 1 course from $S_1 = \{A, B\}$ and 1 course from $S_2 = \{B, C\}$, and you take course B, you may use it to satisfy the first requirement **or** the second requirement, but not both.

   Given a list of requirements $r_1, r_2, ..., r_m$, and a list L of courses taken by a specific student, give a polynomial-time algorithm which determines whether this student can graduate or not.

5. At the mega-corporation Algorithms Inc., there are n employees. There are m total days, and on day i, some subset S_i of the employees are scheduled to work. There are many tasks that must be completed each day involving the implementation and analysis of various algorithms, but the least popular task is the proof of correctness. On day i, a single employee in S_i is selected to do the proof work for that day (this employee must of course be scheduled to work that day). Since the employees would rather not be assigned this specific task, we want the proof work to be divided as fairly as possible.

   Say that person p works on k specific days, and there are $n_1, n_2, ..., n_k$ total employees working each of those days. We will say that person p has been assigned a reasonable amount of proof work if they are assigned the proof task no more than $\lceil \frac{1}{n_1} + \frac{1}{n_2} + ... + \frac{1}{n_k} \rceil$ times. For a simple example where $n = 4$ and $m = 2$, if Adam and Bertha work on day 1, and Adam, Christine, and Dave work on day 2, then it would be unreasonable to have Adam be assigned the proof of correctness both days, since $2 > \lceil \frac{1}{2} + \frac{1}{3} \rceil$, but any other assignment is reasonable.

   Prove that a solution exists which is reasonable for all employees, and give a polynomial-time algorithm to compute such a solution.

   Practice Problems on back.

If you would like some extra practice, you may do the following problems. Do not submit them, as they will not be graded. If you would like to check your answers, talk to the instructor or TA via email or office hours. All extra practice problems are from the Kleinberg and Tardos textbook.

Chapter 7: exercises 8, 12, 14, 17, 31, 41