CSC373S **Problem Set 2** **Spring, 2013**

Due: March 1, beginning of lecture

NOTE: Each problem set only counts 5% of your mark, but it is important to do your own work (but see below). Similar questions will appear on the second term test. You may consult with others concerning the general approach for solving problems on assignments, but you must write up all solutions entirely on your own. Anything else is *plagiarism*, and is subject to the University's Code of Behavior. You will receive 1/5 points for any (non bonus) question/subquestion for which you say "I do not know how to answer this question". You will receive .5/5 points if you just leave the question blank. Advice: Do NOT spend an excessive amount of time on any question and especially not on a bonus question. If you wish to spend "free time" thinking about (say) bonus questions that is fine but you should not sacrifice time needed for other courses.

1. (15 points)

   Consider a flow network $\mathcal{F} = (G, s, t, c)$ with integral capacities that also has a cost function $p : E \rightarrow \mathbf{N}$ associated with the edges. The interpretation of the cost function p is that $p(e)$ is the (non-negative) cost to increase the capacity of edge e by one unit.

   Suppose that we have already computed an optimal flow f for the network $\mathcal{F}$. Now suppose that you want to increase this max flow f by one unit by increasing the capacity of some edges. Our goal is to do this at the least cost.

   Describe a method for optimally determining which edge capacities to increase. Your method should run in time that is asymptotically better than $O(mn)$.

2. (15 points) Consider the following makespan problem: An input $\mathcal{I} = \{I_1, \ldots, I_n\}$ where each job $I_j = (p_j, S_j)$ with p_j the processing time or load of job I_j (as in the makespan problem previously considered in lecture 5) and $S_j \subseteq \{1, 2, \ldots m\}$ is the set of allowable machines on which I_j can be processed. This is called the "restricted machines model". Since this problem extends the "identical machines model" (as in lecture 5), it is also NP hard.

   In what follows, we will only consider the case when all processing times/loads $p_j = 1$. The goal is again to minimize the makespan; that is, minimize the maximum time/load on any machine.

   - (5 points) Show that the "natural online greedy algorithm" (i.e. consider each job in the order given and schedule it on a least loaded machine) is not an optimal algorithm.
   - (bonus) What is the worst approximation you can achieve for the natural online greedy algorithm? Can you construct any greedy algorithm which achieves a constant approximation ratio (independent of n and m)?

- (10 points) Given an integer k, show how you can efficiently determine whether or not the makespan is at most k by reducing the problem to a max flow problem.

3. Show how to polynomial time reduce the 3-colouring search problem to the 3-colouring decision problem. Hint: Introduce three new nodes to represent the three colours and then try to colour one node at a time.

4. Show that the following decision problems are NP-complete. To establish the NP hardness, you can use a polynomial time transformation from any problem in the tree of transformations on slide 14 of lecture 15.

   (a) Half independent set (HIS) where
   HIS $= \{G|G = (V, E)$ and G has an independent set of size at least $\lceil |V|/2 \rceil \}$.

   (b) Unweighted Job Scheduling (UJS) where
   UJS $= \{(\mathcal{I}, k)|\mathcal{I} = \{J_1, \ldots, J_n\}$ is a set of n jobs for which at least k can be scheduled within their deadline without intersection except at the end points$\}$. Here job $J_i = (r_i, d_i, p_i)$ is described by its release time r_i, deadline d_i and processing time p_i. Such a job must be scheduled to start at some time t_i such that $r_i \leq t_i \leq d_i - p_i$ so that it completes by its deadline d_i.

   Hint: Use the subset sum problem.