The Design and Analysis of Algorithms	Problem Set 1
CS 681 Fall 2003	Due Monday, September 15, 2002

(1) (20 points) For each part of this problem give a proof, or a counter-example with explanation.

You may use the following variants of the two main spanning tree lemmas from class without proof.

Lemma 1 *Consider an edge e that is crossing a cut (A, B). If edge e has the unique minimum cost among all edges crossing the cut, then all minimum cost spanning trees contain e. If e is one of the minimum cost edges crossing the cut, then there exists a minimum cost spanning tree that contains e.*

Lemma 2 *Consider an edge e in a cycle C. If edge e has the unique maximum cost among all edges in the cycle, then no minimum cost spanning tree contains edge e. If e is one of the maximum cost edges in the cycle, then there exists a minimum cost spanning tree that does not contain edge e.*

(a.) Consider the minimum spanning tree problem in an undirected graph $G = (V, E)$, with a cost $c_e \geq 0$ on each edge. Assume all edge-costs are different. Suppose you are given a spanning tree T with the guarantee that for every $e \in T$, e belongs to *some* minimum-cost spanning tree in G. Can we conclude that T itself must be a minimum-cost spanning tree in G?

(b.) Consider the same question for the minimum-cost arborescence problem in a directed graph $G = (V, E)$. Suppose you are given an arborescence $A \subseteq E$ with the guarantee that for every $e \in A$, e belongs to *some* minimum-cost arborescence in G. Can we conclude that A itself must be a minimum-cost arborescence in G?

(c.) Consider the special case of the question in part (b.) when G is an acyclic graph, that is, it contains no directed cycles. Can we conclude in this special case that A itself must be a minimum-cost arborescence in G?

(d.) How would your answer to (a), (b), and (c) change if we do not assume that all edge-costs are different?

(2) (10 points) Recall that a matroid is defined by a set of independent set $\mathcal{I}$ of a ground set S that satisfies the following properties:

(i) $\emptyset \in \mathcal{I}$,

(ii) $X \subset Y$ and $Y \in \mathcal{I}$ implies that $X \in \mathcal{I}$,

(iii) $X, Y \in \mathcal{I}$ and $|X| < |Y|$ implies that there exists $y \in Y \setminus X$ such that $X \cup \{y\} \in \mathcal{I}$.

In class we have shown that the greedy algorithm finds the maximum weight independent set in matroids. A closely related definition is that of a greedoid. A *greedoid* is a set system $\mathcal{I}$ that satisfies the above properties (i) and (iii) (without assuming (ii).) For both questions give a proof, or a counter-example with explanation.

(a.) Suppose you use the greedy algorithm from matroids to find an independent set of maximum size (i.e., assume for now that all weights are 1). Does the greedy algorithm find an independent set of maximum size in greedoids?

(b.) Now assume that each element of $s \in S$ has a non-negative weight w_s. Is it true that the greedy algorithm finds a maximum weight independent set in greedoids?