CSE 431
Spring Quarter 2004
Assignment 6
Due Friday, May 21, 2004

All solutions should be neatly written or type set. All major steps in proofs and algorithms must be justified.

1. (10 points) It is well known that the Hamiltonian Path problem is NP-complete (cf. 262-268 of Sipser). The Hamiltonian Path problem is defined by:

Input:	An undirected graph $G = (V, E)$ and vertices $s, t \in V$.
Property:	There is a path in G from s to t that visits every vertex of G exactly once.

   Show that the problem of Bounded Degree Spanning Tree is also NP-complete. Bounded Degree Spanning Tree problem is defined by:

Input:	A connected undirected graph $G = (V, E)$ and number k.
Property:	There is a connected subgraph T of G such that T contains all the vertices of G, contains no cycles, and each vertex of T has degree $\leq k$.

   Part of your proof should be the construction of a polynomial time reduction of Hamiltonian Path to Bounded Degree Spanning Tree.

2. (10 points) Define a $\{\cup, \cdot\}$-regular expression as one that only uses union and concatenation, but not Kleene star. We define the Not Everything problem for $\{\cup, \cdot\}$-regular expressions by:

Input:	A $\{\cup, \cdot\}$-regular expression α over Σ and number k.
Property:	$L(\alpha) \neq \Sigma^k$.

   Show that the Not Everything Problem for $\{\cup, \cdot\}$-Regular Expressions is NP-complete.

   Note that showing the problem is in NP is not altogether trivial. A first step would be to convert α into an NFA. I would recommend showing that 3-CNF-SAT is mapping reducible in polynomial time to this problem. If the CNF formula F uses Boolean variables $x_1, x_2, \ldots, x_n$ then a string in $\{0, 1\}^n$ can represent an assignment to the variables. Try to design a $\{\cup, \cdot\}$-regular expression α over the alphabet $\{0, 1\}$ with the property that $w \in L(\alpha)$ if and only if w is of length n that does not represent a satisfying assignment of F. In other words, $L(\alpha) \neq \{0, 1\}^n$ if and only if F is satisfiable.

3. (10 points) Consider the following scheduling problem with release times and deadlines, called *Job Scheduling*.

   - Input: n jobs of lengths $L_1, \ldots, L_n$, with release times $R_1, \ldots R_n$ and deadlines $D_1, \ldots D_n$, all integers.

- Property: All jobs can be scheduled after their release times, finished before their deadlines, without overlapping. That is, the jobs have start times $S_1, \ldots S_n$, such that, for all i, $R_i \leq S_i$, $S_i + L_i \leq D_i$, and for $j \neq i$, either $S_i + L_i \leq S_j$ or $S_j + L_j \leq S_i$

Show that Job Scheduling is NP-complete. As part of your proof show that Subset Sum is polynomial Time reducible to Job Scheduling.