Homework 1

Problem 1: The *Half 3SAT* problem is defined as follows: We are given a 3CNF formula with n variables and m clauses, where m is even. We wish to determine a truth assignment such that exactly half the clauses are satisfied. Prove that *Half 3SAT* is NP-complete.

Problem 2: Prove that 2SAT can be solved in polynomial time. Make your algorithm as efficient as possible.

Problem 3:

(a) *0-1 Integer Programming*: Given an integer m x n matrix A and an integer m-vector b, the problem asks whether there is an integer n-vector x with elements in $\{0, 1\}$ such that $Ax <= b$. Prove that this problem is NP-complete (Hint: reduce from 3-SAT)

(b) *Integer Programming*: This is the same problem as part (a) but we now relax the restriction on the values of x, and only require that they be integers. Prove that this problem is NP-complete

Problem 4: If all the numbers specified in the job scheduling problem (discussed in class) are specified in unary, give a polynomial time algorithm to solve the problem.

Problem 5: The *Hitting Set* problem: Consider a set $\{a1,\ldots an\}$, and a collection of subsets $B1$, $\ldots Bm$ where each Bi is a subset of A. Given an integer k, we are asked to determine whether there exists yet another subset C of A, such that $|C| <= k$, and C contains at least one element of each Bi (i.e., C "hits" or "touches" each Bi). Prove that the *Hitting Set* problem is NP-complete

Problem 6: The *Undirected Feedback Set* Problem: Given an undirected graph $G = (V, E)$, a *Feedback Set* X is a subset of vertices such that the graph $G - X$ has no cycles. Given k, the problem is to find whether there is a feedback set of size at most k. Prove that this problem is NP-complete.