Problem Set 5 Due: November 10, 2008

Homework: (Total 100 points) Do the following exercises.

Problem 1. [20 points] Prove that the following problem, called 4TA-SAT, is NP-complete. The problem is defined as follows:
INPUT: A boolean formula $F(X_1, X_2, \ldots, X_n)$.
PROBLEM: Does F have at least 4 satisfying truth assignments?

You should reduce the 3-SAT problem to 4TA-SAT. The answers to the following questions constitute the proof of NP-completeness.
(a). Prove that 4TA-SAT is in NP.
(b). Describe a polynomial-time reduction from 3-SAT to 4TA-SAT. Your reduction should take a 3-SAT formula F and construct an instance of the 4TA-SAT problem, say the formula G.
(*Hint*: Suppose you add a variable Y to the boolean formula F, but don't actually use it in any of the clauses. If F had k satisfying truth assignments originally, how many satisfying truth assignments will it have now?)
(c). Show that F is satisfiable if and only if G has at least four satisfying truth assignments.

Problem 2. [20 points] Prove that the following problem called DS (Dominating Set) is NP-complete. A dominating set in a graph $G = (V, E)$ is a set of vertices $S \subseteq V$ such that each vertex in V is either in S or has an edge to some vertex in S. The definition of the problem is the following:
INPUT: Graph $G = (V, E)$ and integer k.
PROBLEM: Does G have a dominating set of size at most k?

We suggest you use the following reduction from 3-SAT to DS. The reduction function takes as input a 3-SAT formula $F(X_1, X_2, \ldots, X_n)$ with clauses $C_1, \ldots, C_m$ such that $C_i = Z_{i1} \vee Z_{i2} \vee Z_{i3}$, where Z_{ij} denotes a literal. The output of the reduction is a graph $G = (V, E)$ and $k = n$. The graph is constructed as follows. For each clause C_i there is a vertex c_i. For each variable X_j there are three vertices x_j, $\overline{x}_j$ and y_j. The three vertices for each variable are connected to each other to form a triangle. The vertex for a clause is connected by an edge to each of the three vertices that correspond to its literals.

Problem 3. [20 points] Given a graph $G = (V, E)$, a *clique* in G is a set of vertices $S \subseteq V$ such that for any pair of distinct vertices $u, v \in S$, the edge (u, v) is present in G. The *size* of the clique is the number of vertices in it. We define CLIQUE to be the following decision problem:
INPUT: A graph $G = (V, E)$, and an integer k.
PROBLEM: Does G have a clique of size at least k?

Let 50-CLIQUE be the following variant on the CLIQUE problem:
INPUT: A graph $G = (V, E)$.
PROBLEM: Does G have a clique of size at least 50?

For two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, we say that G_1 contains a copy of G_2 as a subgraph if there exists a subset $S \subseteq V_1$ of the vertices in G_1 with $|S| = |V_2|$ and a bijection $f : S \to V_2$ such that, for all pairs of vertices $u, v \in S$, $u \neq v$, $(u, v) \in E_1$ if and only if $(f(u), f(v)) \in E_2$. The SUBGRAPH ISOMORPHISM problem is defined as follows:
INPUT: Two graphs G_1 and G_2.
PROBLEM: Does G_1 contain a copy of G_2 as a subgraph?

The CLIQUE problem is NP-complete (see Exercise 10.4.1 in the textbook).
(a). Prove one of the following two statements: (i) 50-CLIQUE is in P; (ii) 50-CLIQUE is NP-hard.
(b). Prove that SUBGRAPH ISOMORPHISM is NP-complete.

Problem 4. [20 points] Prove that the following problem, called TRUE-SAT, is NP-complete. The definition of the problem is the following:
INPUT: A boolean formula $F(X_1, X_2, \ldots, X_n)$ such that $F(T, T, \ldots, T) = T$. In other words, F can be satisfied by setting each variable X_j to TRUE.
PROBLEM: Does F have a satisfying truth assignment in which at least one of the variables is set to FALSE?

We propose the following reduction from 3-SAT to TRUE-SAT. Given a 3-SAT formula F with clauses $C_1, \ldots, C_m$ and the variables $X_1, \ldots, X_n$, we construct a CNF formula G which is an instance of the TRUE-SAT problem. The new formula G has one new variable called Y. For each clause C_i in F, we add the clause $C_i \vee Y$ to G. We also add to G clauses of the form $\overline{Y} \vee X_j$, for each variable X_j. For example, if $F = (X_1 \vee \overline{X}_2 \vee X_3) \wedge (\overline{X}_1 \vee X_2 \vee X_3)$ then $G = (Y \vee X_1 \vee \overline{X}_2 \vee X_3) \wedge (Y \vee \overline{X}_1 \vee X_2 \vee X_3) \wedge (\overline{Y} \vee X_1) \wedge (\overline{Y} \vee X_2) \wedge (\overline{Y} \vee X_3)$.

Problem 5. [20 points] Recall that, in the 3-SAT problem, the input formula must have exactly *three* literals per clause. The 2-SAT problem, which is defined as follows, is the analogue of 3-SAT in which the formula has two literals per clause.
INPUT: A boolean formula $F(X_1, \ldots, X_n)$ in conjuctive normal form with exactly *two* literals per clause.
PROBLEM: Does there exist a satisfying truth assignment for F?

You are required to show that there is a polynomial-time reduction from the 2-SAT problem to the 3-SAT problem.
(a). Provide a polynomial-time reduction which will convert a 2-SAT formula F into a 3-SAT formula G. Briefly explain why your reduction works in polynomial time.
(b). For your reduction, show that the input F is satisfiable if and only if the output G is satisfiable.
(c). Does all this prove that 2-SAT is NP-hard? Explain.