CSE 431 Spring 2015
Assignment #7

Due: Friday, May 29, 2015

Reading assignment: Read Sections 7.5 and 8.1-8.3.

Problems:

1. Let ϕ be a 3CNF-formula. An NOTALLEQUAL assignment to the variables of ϕ is one where each clause contains two literals with different truth values. In other words a NOTALLEQUAL assignment satisfies ϕ but does not set all three literals to true in any clause.

   (a) That the negation of a NOTALLEQUAL assignment for ϕ is also a NOTALLEQUAL assignment for ϕ.

   (b) Let *NAESAT* be the set of all 3CNF formulas ϕ that have a NOTALLEQUAL assignment. Prove that *NAESAT* is NP-complete. For the hardness part use a reduction from 3SAT.
   (Hint: Show that the function that replaces each clause C_i of ϕ of the form $(y_1 \vee y_2 \vee y_3)$ where y_1, y_2, y_3 are literals by the two clauses $(y_1 \vee y_2 \vee z_i)$ and $(\overline{z_i} \vee y_3 \vee w)$ where w is a single new variable for all clauses and there is one z_i variable per original clause.)

2. A *cut* in an undirected graph G is a partition of the vertices V of G into two disjoint parts S and T with $V = S \cup T$. The *size* of the cut (S, T) is the number of edges that cross between S and T. Define

$$MAXCUT = \{\langle G, k\rangle \mid G \text{ has a cut of size } \geq k\}.$$

   Show that $MAXCUT$ is NP-complete. For the hardness part use the fact that $NAESAT$ is NP-complete.
   (Hint: For each variable x in an m-clause 3CNF formula, have $3m$ vertices for x and $3m$ vertices for $\overline{x}$ for a total of $6m$ vertices in the graph. Join each pair of nodes with the same variable label, but different signs by an edge. For each clause, consider a separate vertex of every possible literal label to be dedicated to that clause. Among those literals, add edges that join the 3 literals that actually appear in the clause to form a triangle. What value of k should you use? Prove that this works.)

3. A subset of the nodes of a graph G is a *dominating set* iff every other node of G is adjacent to some node in the subset. Let

$$DOMINATING\text{-}SET = \{\langle G, k\rangle \mid G \text{ has a dominating set with } k \text{ nodes}\}.$$

   Show that $DOMINATING\text{-}SET$ is NP-complete using the NP-completeness of $VERTEX\text{-}COVER$.

4. This problem is inspired by the single-player game *Minesweeper*, generalized to an arbitrary graph. Minesweeper begins with an undirected graph G in which each node either contains a single, hidden mine or is empty. The player chooses nodes, one by one. If the player chooses a node containing a mine, the player loses. If the player chooses an empty node, the player learns the number of neighboring nodes containing mines. The player wins if and when the player has chosen all the empty nodes. (In the actual game it suffices for the player to learn the number of neighboring nodes associated with each empty node.)

   We are interested in the related problem of *Mine-Consistency* in which the input is an undirected graph G together with numbers for some of G's nodes. The goal is to determine whether there is a placement of mines on the remaining nodes so that any node u numbered k has exactly k neighboring nodes containing mines. Formulate *Mine-Consistency* as a language, and prove that it is NP-complete.

   Hint: One possibility is a reduction from 3SAT. The reduction from 3SAT to SUBSET-SUM in the text might inspire you in the right direction.

5. (Bonus) Sipser's text: Problem 9.25 2nd Edition (Problem 9.16 1st Edition, Problem 9.24 3rd Edition).

6. (Bonus) Sipser's text: Problem 9.26 2nd Edition (Problem 9.17 1st Edition, Problem 9.25 3rd Edition).