Theory of Computation

homework 3
Due: 11/19/2013

**Problem 1** Prove that the following language is coNP-complete.

\[ L_{\text{coNP}} = \{ \phi : \text{a Boolean formula that is satisfied by every assignment} \} \]

**Ans:** It is clear that \( L_{\text{coNP}} \) is in coNP by its definition. We then prove that every \( L \in \text{coNP} \) can be reduced to \( L_{\text{coNP}} \). First, we know that \( \bar{L} \) (which is in NP) can reduce to SAT (an NP-complete problem). For every input \( x \in \{0,1\}^* \) that reduction produces a formula \( \phi_x \) that is satisfiable iff \( x \in \bar{L} \). On p. 424 of the lecture notes, we know that \( L' \) is coNP-complete iff \( \bar{L'} \) is NP-complete. Hence SAT COMPLEMENT is coNP-complete and \( L \in \text{coNP} \) can reduce to SAT COMPLEMENT. As \( \phi_x \) is unsatisfiable iff \( x \in L \), we can readily see that the *same* reduction shows that \( L_{\text{coNP}} \) is coNP-complete.

**Problem 2** Given a set \( S = \{a_1, a_2, ..., a_n\} \) and a number \( T \), we ask if there exists a subset \( S' \subseteq S \) such that \( \sum_{a_i \in S'} a_i = T \). Prove that this problem is NP-complete.

**Ans:** An instance of KNAPSACK contains \( n \) items with values \( v_1, ..., v_n \) and weights \( w_1, ..., w_n \), a weight limit \( W \), and a goal \( K \). KNAPSACK asks if there exists a subset \( S \subseteq \{1,2,...,n\} \) such that \( \sum_{i \in S} w_i \leq W \) and \( \sum_{i \in S} v_i \geq K \). We now reduce KNAPSACK to our problem by simply letting \( x_i = 0,1 \), \( w_i = v_i \) and \( W = K \) to give us the equation \( \sum_{i \in S} w_i x_i = K \). Clearly, a solution to this instance exists if and only if a solution \( S \) exists such that \( \sum_{a_i \in S'} a_i = T \). Since this version of KNAPSACK is NP-complete (refers to slide p. 393), our problem is hence NP-complete.