CS180 Homework 9

Due: 4:00pm, 06/01/2016

1. A 2-CNF stands for a conjunction normal form boolean formula in which each clause consists 2 literals. For example, the formula $(x_1 \vee x_2) \wedge (x_3 \vee \neg x_2)$ is a 2-CNF. The **2-SAT** problems asks given a 2-CNF formula φ, decide whether it is satisfiable or not. Give a polynomial-time algorithm for **2-SAT**:

   (a) The boolean formula $p \Rightarrow q$ is false only when $p = \texttt{True}$ and $q = \texttt{False}$. Show that $(x_1 \vee x_2)$ iff $(\neg x_1 \Rightarrow x_2) \vee (\neg x_2 \Rightarrow x_1)$.

   (b) Show that if $(p \Rightarrow q) \wedge (q \Rightarrow r)$, then if the formula is evaluated for True when $p = \texttt{True}$ then it must be that $r = \texttt{True}$.

   (c) Think of the implication relation is an edge between two literals and build an implication directed graph G for the boolean formula φ.

   (d) Prove that φ is satisfied iff for all x_i we have that x_i and $\neg x_i$ are not in the same strongly connected component of G.

   (e) Use an algorithm to detect a strongly connect component in the implication graph to solve the **2-SAT** problem.

   (f) If φ is satisfiable, find a satisfying assigment.

2. Given a weighted graph $G = (V, E)$ and an integer k. The Longest Path problem asks whether there exists a simple path of length greater or equal to k. The Shortest Path problem asks whether there exists a simple path of length smaller or equal to k.

   (a) Show a polynomial reduction from the Hamiltonian Path problem to the Longest Path problem.

   (b) Show a polynomial reduction from the Hamiltonian Path problem to the Shortest Path problem.

3. Show that the following three problems are polynomial time reducible to each other.

   - **Set-Cover**: Given a collection of sets, and a number k, the Set-Cover problem asks if there are at most k sets from the collection of sets such that their union contains every element in the union of all sets.
   - **Hitting-Set**: Given a collection of sets, and a number k, the Hitting-Set problem asks if are there at most k elements of the sets such that there is at least one element from each set? (PS: In class Prof. Gafni actually proved the Hitting-Set to be NP-complete, rather than Set-Cover. In this problem you see that Set-Cover is also therefore NP-complete)
   - **Dominating-Set**: Given an undirected graph G, and a number k, the Dominating-Set problem asks if there is a subset of vertices of size $\leq k$ such that every vertex in graph is either in the subset or has a neighbor that is in the subset.

(a) Show a polynomial reduction from **Hitting-Set** to **Dominating-Set**.

(b) Show a polynomial reduction from **Dominating-Set** to **Hitting-Set**.

(c) Show a polynomial reduction from **Set-Cover** and **Dominating-Set**.

(d) Show a polynomial reduction from **Dominating-Set** to **Set-Cover**.

4. The degree bounded spanning tree problem is given by a undirected graph $G = (V, E)$, and a number k. The problem is to decide whether or not G has a spanning tree T such that each vertex in T has a degree of at most k. Prove that the degree bounded Spanning Tree is NP-hard. (Hint: reduce a well known NP-complete problem you have seen to the degree bounded Spanning Tree problem)

★ Homework assignments are **STRICTLY** due on the exact time indicated. Please submit a hard copy of your homework solution with your **Name**, **Bruin ID**, **Discussion Number**, clearly indicated on the first page. If your homework consists of multiple pages, please **staple** them together. Email attachments or other electronic delivery methods are not acceptable.

★ We recommend to use LaTeX, LyX or other word processing software for submitting the homework. This is not a requirement but it helps us to grade the homework and give feedback. For grading, we will take into account both the correctness and the clarity. Your answer are supposed to be in a simple and understandable manner. Sloppy answers are expected to receiver fewer points.

★ Unless specified, you should justify your algorithm with proof of correctness and time complexity.