COMPSCI330 Design and Analysis of Algorithms
Assignment 10

Due Date: Friday, Dec. 8, 2017

Guidelines

- **Describing Algorithms** If you are asked to provide an algorithm, you should clearly define each step of the procedure, and then analyze its overall running time. There is no need to write pseudo-code; an unambiguous description of your algorithm in plain text will suffice. If the running time of your algorithm is worse than the suggested running time, you might receive partial credits.
- **Typesetting and Submission** Please submit each problem as an *individual* pdf file for the correct problem on Sakai. LaTeX is preferred, but answers typed with other software and converted to pdf is also accepted. Please make sure you submit to the correct problem, and your file can be opened by standard pdf reader. **Handwritten answers or pdf files that cannot be opened will not be graded.**
- **Timing** Please start early. The problems are difficult and they can take hours to solve. The time you spend on finding the proof can be much longer than the time to write. If you submit within one week of the deadline you will get half credit. Any submission after that will not receive any credit.
- **Collaboration Policy** Please check this page for the collaboration policy. You are **not** allowed to discuss homework problems in groups of more than 3 students. **Failure to adhere to these guidelines will be promptly reported to the relevant authority without exception.**

Problem 1 (Vertex Cover)**.** In the VERTEX-COVER problem, your input is a (undirected) graph G and an integer k. We call a set of vertices S a vertex cover if every edge in G has at least one endpoint in S. The VERTEX-COVER problem asks you to decide whether the graph G has a vertex cover of size at most k.
(a) (10 points) Reduce INDEPENDENT-SET to VERTEX-COVER.
(Hint: the number k in VERTEX-COVER and INDPENEDENT-SET need not be the same.)
(b) (15 points) In an alternative version of VERTEX-COVER, which we call HALF-COVER, the input is just a graph G with $2n$ vertices, and you need to decide whether G has a vertex cover of size at most n. Use reduction to show HALF-COVER is also NP-hard.

Problem 2 (Bounded Tree)**.** (20 points) Given an undirected graph and vertices s, t, a hamiltonian path from s to t is a path from s to t that visit each vertex *exactly* once. The HAMILTONIAN PATH problem asks you to decide whether there is a hamiltonian path from s to t. HAMILTONIAN PATH is a well-known NP-complete problem.

Now we consider a different problem called BOUNDED TREE, given a graph, BOUNDED TREE asks you to decide whether the graph has a spanning tree where every node is adjacent to at most 3 edges in the spanning tree. Reduce HAMILTONIAN PATH problem to BOUNDED TREE problem to show that BOUNDED TREE problem is NP-hard.

(Hint: You need to add some vertices/edges so that a hamiltonian path becomes a part of a spanning tree.)

Problem 3 (Dominating Set)**.** (25 points) Remember Problem 3 in homework 5 where we were trying to find locations for convenient stores? Now we want to open fewer stores and still achieve the same guarantee. Just as a recap: there are n neighborhoods in North Carolina. You are given a map where neighborhoods are vertices and adjacent neighborhoods are connected by an edge. Suppose we would like to open some stores in North Carolina, so that everyone can either go to a store in his/her neighborhood, or a store in an adjacent neighborhood. Now the STORE problem asks you to decide whether it is possible to open at most k stores (k is an integer given as part of the input) that satisfies the requirement – everyone can find a store either in their neighborhood or in at least one of the adjacent neighborhoods.

Based on the fact that VERTEX-COVER is NP-complete, show that the STORE problem is also NP-complete. (Hint: There are several ways to do this, but they all involve constructing a new graph based on the original graph.)