COS 423	Theory of Algorithms	Spring 2018

Problem Set 8

This assignment is due at 5pm on Tuesday, May 15 (Dean's Date) via electronic submission. Collaboration is **not** *permitted, except with the course instructor and two preceptors. No extensions or late days (without the recommendation of a Dean).*

Read CHAPTER 8 and SECTION 10.2 in *Algorithm Design*.

1. Consider the following decision and optimization versions of the longest path problem:

   - LONGEST-PATH: Given an undirected G with integer edge weights $w(e) \geq 1$ and an integer L, *does there exist* a simple path (no repeated nodes) whose length is $\geq L$?
   - FIND-LONGEST-PATH: Given an undirected graph G with integer edge weights $w(e) \geq 1$, *find* a longest simple path.

   Prove that FIND-LONGEST-PATH $\equiv_{\mathrm{P}}$ LONGEST-PATH.

2. Consider the following two related problems:

   - SUBSET-SUM: Given n natural numbers $w_1, \ldots, w_n$ and an integer W, is there a subset that adds up to exactly W? A subset may contain each number at most once.
   - COIN-CHANGING: Given m coin denominations $1 = c_1 < \ldots < c_m$ and an amount S, can you make change for the amount S using at most T coins? You may use as many coins of each coin denomination as desired.

   (a) Prove that SUBSET-SUM $\leq_{\mathrm{P}}$ COIN-CHANGING.
   Hint: as in the reduction from 3-SAT *to* SUBSET-SUM*, use the individual digits of the* COIN-CHANGING *instance to impose any desired constraints (e.g., that you will take at most one coin of each denomination). Express the digits in base b for a value of b that is sufficiently large that there are no carries.*

   (b) Prove that COIN-CHANGING is **NP**-complete.

3. Design a linear-time algorithm for FIND-LONGEST-PATH (defined above) when G is a tree.