CS 325 - Homework 7

1. *(7 pts)* Let X and Y be two decision problems. Suppose we know that X reduces to Y in polynomial time. Which of the following can we infer? Explain

   a. If Y is NP-complete then so is X.
   b. If X is NP-complete then so is Y.
   c. If Y is NP-complete and X is in NP then X is NP-complete.
   d. If X is NP-complete and Y is in NP then Y is NP-complete.
   e. X and Y can't both be NP-complete.
   f. If X is in P, then Y is in P.
   g. If Y is in P, then X is in P.

2. *(4 pts)* Consider the problem COMPOSITE: given an integer *y*, does *y* have any factors other than one and itself? For this exercise, you may assume that COMPOSITE is in NP, and you will be comparing it to the well-known NP-complete problem SUBSET-SUM: given a set S of *n* integers and an integer target *t*, is there a subset of *S* whose sum is exactly *t*? Clearly explain whether or not each of the following statements follows from that fact that COMPOSITE is in NP and SUBSET-SUM is NP-complete:

   a. SUBSET-SUM $\leq_p$ COMPOSITE.
   b. If there is an $O(n^3)$ algorithm for SUBSET-SUM, then there is a polynomial time algorithm for COMPOSITE.
   c. If there is a polynomial algorithm for COMPOSITE, then P = NP.
   d. If P $\neq$ NP, then ***no*** problem in NP can be solved in polynomial time.

3. *(3 pts)* Two well-known NP-complete problems are 3-SAT and TSP, the traveling salesman problem. The 2-SAT problem is a SAT variant in which each clause contains at most two literals. 2-SAT is known to have a polynomial-time algorithm. Is each of the following statements true or false? Justify your answer.

   a. 3-SAT $\leq_p$ TSP.
   b. If P $\neq$ NP, then 3-SAT $\leq_p$ 2-SAT.
   c. If P $\neq$ NP, then no NP-complete problem can be solved in polynomial time.

4. *(6 pts)* A Hamiltonian path in a graph is a simple path that visits every vertex exactly once. Show that HAM-PATH = { (G, u, v): there is a Hamiltonian path from u to v in G} is NP-complete. You may use the fact that HAM-CYCLE is NP-complete

5. *(5 pts)* LONG-PATH is the problem of, given (G, u, v, k) where G is a graph, u and v vertices and k an integer, determining if there is a simple path in G from u to v of length at least k. Prove that LONG-PATH is NP-complete.

EXTRA CREDIT *(5 pts)*

The **Traveling Purchaser Problem** (**TPP**):

The travelling purchaser travels from marketplace to marketplace searching for goods on his list. The travelling purchaser always returns to his "home" marketplace with all the goods on his list and having spent as little money as possible.

Given a list of marketplaces, the cost of travelling between different marketplaces, and a list of available goods together with the price of each such good at each marketplace, the task is to find for a given "shopping list" of goods the purchasing route with the minimum combined cost of purchases and traveling. The purchasing route solution will indicate both the order the marketplaces are visited and the goods that are purchased at each marketplace. The purchaser must start and finish at the same marketplace and visit each other marketplace at most once (it is not required that all marketplaces be visited).

The decision version of **TPP-D** would ask if there exists a route for a set of marketplaces, travel costs, list of goods, and costs of goods such that the total cost of purchases and travelling is at most k.

Formally **TPP-D** = { (G, C, L, A, P, k) : where G=(V,E) where the vertices are marketplaces, E is the set of edges which represent the road between the marketplaces, C(u,v) is the cost of travelling from marketplace u to v, L is the shopping list of goods that the purchaser wants to buy, A(v) is the list of goods available at marketplace v and P(x,v) is the price of the good x at marketplace v. G has a TPP route with total cost at most k }.

Prove that **TPP-D** is NP-Complete