Stephen Townsend
CS 325-400
11/19/17

# Assignment 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.

This is not true because X can be NP.

b. If X is NP-complete then so is Y.

This is not true because Y could be NP Hard or something other than NP.

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.

This is correct because NP complete can be reduced to NP and this would show that Y is at least as hard as X being 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.

We can infer d & g. The reason is because if X reduces to Y in polynomial time then we can solve X quickly if we know how to solve Y quickly or efficiently. This makes it to where Y is going to be at least as hard as X.

Stephen Townsend
CS 325-400
11/19/17

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 ≤p COMPOSITE.

   This is false, we know that SUBSET-SUM is NP-Complete and COMPOSITE is NP. Being NP-Complete it can be reduced to another NP Complete problem. Given the information above we can only determine that COMPOSIT is NP and not NP Complete. As a result, it can not be said that SUBSET-SUM can be reduced to COMPOSITE.

b. If there is an $O(n^3)$ algorithm for SUBSET-SUM, then there is a polynomial time algorithm for COMPOSITE.

   This is true, given SUBSET-SUM is NP-complete and it can be solved in polynomial time. This means that all NP problems are solvable in polynomial time.

c. If there is a polynomial algorithm for COMPOSITE, then P = NP.

   This is false, all that is stated is that COMPOSITE is NP. Thus, it is unknown if it is in fact NP-Complete. With that in mind, having a polynomial time algorithm and being NP is not sufficient to imply that P=NP.

d. If P ≠ NP, then no problem in NP can be solved in polynomial time.

   This is false, P ≠ NP shows that NP-Complete problems cannot be solved in polynomial time. The reason is because P is a subset of NP and COMPOSITE in in NP.

Stephen Townsend
CS 325-400
11/19/17

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.

This is true. The reason being is that there was a slide in the NP-complete lecture that indicated 3-Sat has the ability to be reduced to DIR-HAM-CYCLE. That can be further reduced to HAM-CYCLE. From there we are able to reduce a HAM-CYCLE into TSP. Thus, 3-SAT and be reduced into TSP.

b. If P ≠ NP, then 3-SAT $\leq_p$ 2-SAT.

This is false. The notes above state that 2-SAT has a polynomial time algorithm and 3-SAT is NP complete. 2-SAT is within P and if we are able to reduce 3-SAT to 2-SAT then it would be within NP-complete and within P. Thus P = NP.

c. If P ≠ NP, then no NP-complete problem can be solved in polynomial time.

This is true. The reason is because if we are able to solve one NP-Complete problem in polynomial time then we can solve all NP-complete problems in polynomial time as well.

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

We are given that a HAM-CYCLE is NP-Complete. Given, A HAM-Cycle is a HAM-Path that is a cycle. Being a cycle, it is a closed system where the vertices and edges cannot repeat. Moreover, if a HAM-Cycle can be reduced to a HAM-Path, this means not only that a HAM-Path would need to be within NP to start, but to be able to reduce, A HAM-Path is in fact at least as hard as a HAM-Cycle. This would thus make a HAM-Path NP Complete given HAM-Cycle can be reduced and is NP-Complete.

Stephen Townsend
CS 325-400
11/19/17

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.

Under the assumption that LONG-PATH is NP, it would have to follow yes, no decisions. These decisions can be check in polynomial time.

To show that LONG-PATH can be solved in polynomial time, if we have a sequence of vertices that make up the path, it can be transverse in O(n) time. We can then check the adjacency to see if the vertex is within reach. This would show that the problem is at least within NP.

To prove NP-Complete from NP, we can solve the graph with a HAM-CYCLE. A HAM-CYCLE can be reduced into a HAM-PATH. A Ham-Path can be reduced into a LONG-PATH. And given the logic that if you can solve one NP-Complete problem in polynomial time then we can solve all NP-complete problems in polynomial time. We are left with NP-complete (Y) ≤p NP (X) (HAM-CYCLE ≤p LONG PATH).

If Y is NP-complete, and

1. X is in NP

2. Y ≤p X

then X is NP-complete.

Thus LONG-PATH is NP-Complete.