George Lenz
8/12/2018
HW-7

1. Let X and Y be two decision problems. Suppose we know that X reduces to Yin polynomial time. Which of the following can we infer? Explain

a. If Y is NP-complete then so is X.
(FALSE, you still need to prove that the solution can be confirmed in polynomial time)

b. If X is NP-complete then so is Y.
(FALSE, you have proven x is no harder than y, not that y is no harder than X, also you need to prove the solution in polynomial time)

c. If Y is NP-complete and X is in NP then X is NP-complete.
(FALSE, you need to prove that a solution to X can be confirmed in polynomial time)

d. If X is NP-complete and Y is in NP then Y is NP-complete.
(TRUE, if X is NP-complete and X is no harder than y, and x can be reduced to y then y must be NP-complete)

e. If X is in P, then Y is in P.
(FALSE, X is no harder than Y, but Y can be much harder than X)

f. If Y is in P, then X is in P.
(TRUE, if X is no harder than P and Y is in P then X must be in P)

2. 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.

- A Hamiltonian path can be made into a Hamiltonian cycle by adding an edge from the last vertex in a Hamiltonian path connected to the first vertex, therefore creating a Hamiltonian cycle. If the graph is a complete Hamiltonian cycle then it must be a Hamiltonian path, making it np-hard. To prove the solution you can just make sure all the vertexes are used and only used once which can be proven in polynomial time therefore a Hamiltonian path must be np-complete.

3. 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. Show that LONG-PATH is NP-complete.

- A solution to long-path can be proven by adding up the distances to each vertex and comparing to k which can be done in polynomial time. Long-path can be reduced to a Hamiltonian path of the sub graph of the graph G from the path taken from u to v, and in a hamilonian path every vertex must be used only once, therefore if every vertex in the subgraph is used only once, which makes it np-complete.

4. K-COLOR. Given a graph G = (V,E), a k-coloring is a function c: V -> {1, 2, ... , k} such that c(u) c(v) for every edge (u,v) E. In other words the number 1, 2, .., k represent the k colors and adjacent vertices must have different colors.The decision problems K-COLOR asks if a graph can be colored with at most K colors.

a. The 2-COLOR decision problem is in P. Describe an efficient algorithm to determine if a graph has a 2-coloring. What is the running time of your algorithm?

For the 2 color problem you can start at a vertex, and every vertex connected by an edge must be of a different color, so you can color it the opposite color, and keep going for all the vertexes, an .

b. The 3-COLOR decision problem is NP-complete by using a reduction from SAT. Use the fact that 3-COLOR is NP-complete to prove that 4-COLOR is NP-complete.

Each color can be deduced to a value x1 or x2 or x3 and can be connected to each other value based on what it could be. For examble if there were four vertices and three colors, red, green, and blue then whatever color the first is would have to be true and the rest false, from there the next vertices have to be not the first color, and either the second or third color, then depending on which is true or false for the second, you can develop a true or false for the third, and so on for each option. Therefore it is reduceable to SAT. Proving can be done in polynomial time as well, as you could just go to each vertex and check that every vertex attached is of a different color.

4-color can reduce to 3-color because if a graph has 3-color, and you just add one more vertex attached to all the rest of a different color making it 4-color. Therefore both 3-color and 4-color are both np-complete.