Junhyeok Jeong

CS 325 – 400 F2019

November 24, 2019

Homework 7

1.

Conditions

- both X and Y are decision problems.

- X reduces to Y in polynomial time.

a) If Y is NP-complete then so is X.

- This is False. If Y is NP-complete, then X is in NP when X reduces to Y in polynomial time. However, I can't infer the X is the NP-complete because it should be the subset of NP-complete, not NP.

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

- This is False. If X is NP-complete and reduces to Y in polynomial time, then Y can't be NP-complete because Y doesn't reduce to Y in polynomial time.

c) If Y is NP-complete and X is in NP then X is NP-complete.

- This is False. X can be effectively solved by Y because X reduces to Y in polynomial time. This can't mean that X is NP-complete.

d) If X is NP-complete and Y is in NP then Y is NP-complete.

- This is True. Since X reduces to Y in polynomial time, X can be solved easier than Y. Therefore, Y can be NP-complete.

e) If X is in P, then Y is in P.

- This is false. Since X reduces to Y in polynomial time, if Y is in P, then X is in P.

f) If Y is in P, then X is in P.

- This is True. Since X reduces to Y in polynomial time, if Y is in P, then X is in P.

2.

Condition

- COMPOSITE is in NP and SUBSET-SUM is NP-complete

a) This statement doesn't follow. Since SUBSET-SUM is NP-complete, it would be reduced to other NP-complete problem, not NP problem which is COMPOSITE.

b) This statement follows because $O(n^3)$ algorithm means that polynomial time, so polynomial time in NP-complete which is SUBSET-SUM can be interpreted P=NP. Therefore, there is polynomial time algorithm in the NP which is COMPOSITE.

c) This statement doesn't follow because polynomial time algorithm in NP (COMPOSITE) doesn't imply COMPOSITE problem is in NP-complete. This statement should be flipped to follow.

d) This statement doesn't follow because P is a subset of NP. In other words, some problems in NP aren't in P. Therefore, some problems in NP can also be solved in polynomial time.

3.

If Hamiltonian path in the graph G is NP-complete, it should be proved that Hamiltonian path is also in NP and there is NP-complete problem which is can be reduced to Hamiltonian path.

Prove 1) Hamiltonian path is in NP

- From the algorithm logic of Hamiltonian path, each vertex exactly visited once. Therefore, I can verify in polynomial time that Hamiltonian path is simply made of n vertices visits and checking the adjacent n vertices.
- Therefore, Hamiltonian path is in NP.

Prove 2) Find NP-complete problem can be reduced to Hamiltonian path.

- I already know a given fact that HAM-CYCLE is NP-complete. Since it has similar structure to Hamiltonian path, I need to prove that HAM-CYCLE is reduced to Hamiltonian path.
- To prove HAM-CYCLE is reduced to Hamiltonian path, let me add one extra node E in the HAM-CYCLE. Thus, HAM-CYCLE is A-B-C-D-E-A when Hamiltonian path is A-B-C-D. This shows that there is common path A-B-C-D in both HAM-CYCLE and Hamiltonian path.
- Therefore, this proved that HAM-CYCLE is reduced to Hamiltonian path through common path whether adding extra node E or not. In other words, Hamiltonian Path is NP-complete because HAM-CYCLE (NP-complete problem) is reduced to Hamiltonian path.

4.

a) To determine if a graph has a 2-coloring, BFS algorithm can be efficient algorithm because BFS search neighbor vertices first. Therefore, it can be used for distinguishing neighbor vertex has different color or same color.

Algorithm can be described like this:

1. Let graph G = (V, E) and Color array C = [color 1, color 2]
2. Start looping
3. Assign color 1 at even level vertices
4. Check the input vertex's neighbors color.
5. If the color is same with input vertex color, then return false and break loop.
6. If different color, then return true and keep looping.
7. Assign color 2 at odd level vertices.
8. If the color is same with input vertex color, then return false and break loop.
9. If different color, then return true and keep looping.
10. Loop until reached last vertex.

This algorithm is based on BFS algorithm. Therefore, the running time is O(V+E) when V is the number of vertices and E is the number of edges in the Graph G.

b) At first, I know the fact that 3-COLOR decision problem is NP-complete. To prove 4-COLOR is NP-complete, I should show 4-COLOR is in NP and 3-COLOR is reduced to 4-COLOR.

Prove 1) 4-COLOR is in NP

- Let graph G = (V, E) and Color C is mapped from V to {1, 2, 3, 4}.
- We should check the color of each vertex in V is different with the colors of neighbor vertices.
- If one of the neighbor vertices has the same color with input vertex, then this is not 4-COLOR.
- The searching processing time of 4-COLOR can be solved in polynomial time in n, which is the number of vertices and n is at least 4 as 4-COLOR condition.
- Therefore, 4-COLOR is in NP.

Prove 2) 3-COLOR is reduced to 4-COLOR.

- Let graph G is for 3-COLOR and graph G-prime is for 4-COLOR.
- This problem can be proved like HAM-CYCLE is reduced to Hamiltonian path.
- In G graph which is 3-COLORed graph, add one extra vertex to the end of graph with new color. The added vertex graph is called G-prime graph since it has 4 colors.
- In the G-prime graph, it is not violated the rule (adjacent vertices must have different colors) because the new vertex has different color with adjacent vertices.
- This can be reduced to 3-COLOR by delete the new vertex.
- Therefore, 3-COLOR is reduced to 4-COLOR.

Through Prove 1 and 2, the 4-COLOR is NP-complete because 4-COLOR is in NP and 3-COLOR is reduced to 4-COLOR.