Xiaoying Li
CS 325 -Winter 2020
Homework #7

- **Problem 1**

The definition of problem Y is NP-complete is: $Y \in NP\ and\ X \leq_p Y\ for\ all\ X \in NP$. Hence:

<u>a. False.</u>

Because $Y \in NP - complete\ and\ X \leq_p Y$ can only infer $X \in NP$, but not $X \in NP - complete$.

<u>b. False.</u>

Because for NP-complete, Y needs to be the subset of NP, but $X \leq_p Y\ and\ X \in NP - complete$ cannot infer $Y \in NP$.

<u>c. False.</u>

Because $X \leq_p Y$ not $Y \leq_p X$, so Y is NP-complete and X is in NP cannot infer X is NP-complete.

<u>d. True.</u>

Because if one NP-complete problem X is reducible to Y in polynomial time, then all NP-complete problems are reducible to Y in polynomial time, so if X is NP-complete, $X \leq_p Y$ and Y is in NP then Y is NP-complete.

<u>e. False.</u>

Since part d is true, which means when X is NP-complete, Y is in NP and $X \leq_p Y$ then Y is NP-complete is true, we can infer that X and Y can both be NP-complete.

X reduces to Y in polynomial time means that any instance of X can be "easily rephrased" as an instance of Y, the solution to Y provides a solution to the instance of X, means X is "no harder to solve" than Y. Hence:

<u>f. False.</u>

Because X is "no harder to solve" than Y, so $X \in P$ cannot infer $Y \in P$.

<u>g. True.</u>

Because X is "no harder to solve" than Y, so $Y \in P$ can infer $X \in P$.

Therefore, only statements d and g can be inferred.

- **Problem 2**

a. No.

COMPOSITE is in NP and SUBSET-SUM is NP-complete cannot infer $SUBSET - SUM \leq_p COMPOSITE$, since a NP-complete problem does not necessarily reduce to a NP problem.

b. Yes.

Because COMPOSITE is in NP and SUBSET-SUM is NP-complete, and NP problem is no harder than NP-complete problem, so there is a polynomial algorithm for SUBSET-SUM can infer that there is a polynomial time algorithm for COMPOSITE.

c. No.

Because COMPOSITE is only in NP but not in NP-complete, but to infer P=NP, COMPOSITE has to also in NP-complete, so P=NP cannot be inferred.

d. No.

P$\neq$NP means not all problems in NP can be solved in polynomial time, does not mean all problems in NP cannot be solved in polynomial time, there are some problems can be solved in polynomial time. So, P$\neq$NP cannot infer no problem in NP can be solved in polynomial time.

- **Problem 3**

a. True.

Because 3-SAT is NP-complete problem, and all NP-complete problems are NP problems, so 3-SAT$\in$NP. And TSP is NP-complete problem, so all problems in NP reduce to it. Therefore, 3-SAT reduces to TSP, aka $3 - SAT \leq_p TSP$.

b. False.

2-SAT has a polynomial-time algorithm. If $3 - SAT \leq_p 2 - SAT$, then 3-SAT is no harder than 2-SAT and can be solved in polynomial time. And 3-SAT is also NP-complete problem, which indicates that P=NP, which contradicts the supposition that P$\neq$NP. So, the statement is false.

c. True.

Because P$\neq$NP means at least one problem in NP cannot be solved in polynomial time, but if any NP-complete problem can be solved in polynomial time, since all NP problems can be reduced in polynomial time to every NP-complete problem, then all NP problems can be solved in polynomial time, which contradicts the supposition that P $\neq$ NP. So, if P $\neq$ NP, then no NP-complete problem can be solved in polynomial time.

- **Problem 4**

To show HAM-PATH is NP-complete, we first need to show that it belongs to NP, and then show that there is some known NP-complete problem that can be reduced to HAM-PATH in polynomial time.

(1) Show that HAM-PATH $\in$ NP:

For a given solution to a HAM-PATH = {(G, u, v)}, it takes O(v) time to verify if it's a Hamiltonian path from u to v in G, which is obviously in polynomial time. So, HAM-PATH $\in$ NP.

(2) Show that $R \leq_p HAM - PATH$ for some R $\in$ NP-complete:

a. Select R= HAM-CYCLE, because it has a similar structure to HAM-PATH.

A Hamiltonian cycle is a Hamiltonian path that begins and ends in the same vertex, and HAM-CYCLE is NP-complete. If we can show that HAM-CYCLE can be reduced to HAM-PATH in polynomial time, $R \leq_p HAM - PATH$ for some R $\in$ NP-complete is proved.

b. Show a polynomial algorithm to transform HAM-CYCLE into an instance of HAM-PATH.

Given a graph G = (V, E), and construct a graph G': make a copy of G first; then choose an arbitrary vertex u in G and add a copy of u, named u', and all u's edges to G'; then add vertices v and v' to G', and connect v with u and v' with u'. This transformation of G into G' can be done in polynomial time by adding 3 vertices and (number of u's edges+2) edges.

c. Prove we are able to "solve" HAM-CYCLE by using HAM-PATH. Therefore HAM-PATH is as hard as HAM-CYCLE.

Show that G contains a Hamiltonian cycle if and only if G' contains a Hamiltonian path.

First suppose that G contains a Hamiltonian cycle from u to u, thus we can get a Hamiltonian path in G' from v to v': v → follow the Hamiltonian cycle from u back to u' instead of u → v', since v is only connected to u and v' is only connected to u'. So, we proved that if G contains a Hamiltonian cycle, then G' contains a Hamiltonian path.

Then suppose that G' contains a Hamiltonian path from v to v', then if we ignore v and v', the path is from u to u' since v is only connected to u and v' is only connected to u'. Now remove u' and all edges of u', the graph is transformed to G and if we close the path back to u, this path will be a Hamiltonian cycle in G form u to u. Thus, we proved that if G' contains a Hamiltonian path, then G contains a Hamiltonian cycle.

Hence, we proved that G contains a Hamiltonian cycle if and only if G' contains a

Hamiltonian path.

d. Since HAM-CYCLE is in NP-Complete then HAM-PATH must be in NP-Hard.

Since (1) and (2) hold, HAM-PATH is NP-complete.

- **Problem 5**

To show LONG-PATH is NP-complete, we first need to show that it belongs to NP, and then show that there is some known NP-complete problem that can be reduced to LONG-PATH in polynomial time.

(1) Show that LONG-PATH ∈ NP:

For a given solution to a LONG-PATH = (G, u, v, k), we can simply verify it by going through every vertex in the solution to see if every vertex is visited exactly once and the length of the path is at least k in O(v) time, which is obviously polynomial time. So, LONG-PATH ∈ NP.

(2) Show that $R \leq_p LONG - PATH$ for some R ∈ NP-complete:

a. Select R= HAM-PATH, because it has a similar structure to LONG-PATH.

From problem 4 we know that a Hamiltonian path in a graph is a simple path that visits every vertex exactly once, and HAM-PATH is NP-complete. If we can show that HAM-PATH can be reduced to LONG-PATH in polynomial time, $R \leq_p LONG - PATH$ for some R ∈ NP-complete is proved.

b. Show a polynomial algorithm to transform HAM-PATH into an instance of LONG-PATH.

c. Prove we are able to "solve" HAM-PATH by using LONG-PATH. Therefore LONG-PATH is as hard as HAM-PATH.

Given a graph G = (V, E), suppose G has a Hamiltonian path from u to v, by specifying k=number of vertices-1, graph G also has a Long path from u to v of length at least k. Conversely, suppose G has a Long path from u to v of length at least k, where k=number of vertices-1, obviously G also has a Hamiltonian path from u to v.

Thus, we proved that graph G contains a Hamiltonian path if and only if G contains a Long path. And obviously the transformation from HAM-Path to LONG-PATH is polynomial by specifying k=number of vertices-1.

d. Since HAM-PATH is in NP-Complete then LONG-PATH must be in NP-Hard.

Since (1) and (2) hold, LONG-PATH is NP-complete.