CSCI 570. Homework #10.

Boyang Xiao. 3326730274.

Problem # 1

STEP 1: Prove 3-SAT (15/16) is NP problem.

- It has an efficient certificate: A certificate t is an assignment of truth values to variables (x_i).
- It has an efficient certifier: Identify the whole number of clauses that are true can exceed 15/16 of the total number of clauses.

Q.E.D. It is a NP Problem.

STEP 2: We choose a NP-complete problem: 3-SAT (1).

STEP 3: We show 3-SAT (1) $\leq$ 3-SAT (15/16).

3-SAT (1) has 8 possible clauses:

$(x_1 \wedge x_2 \wedge x_3)$
$(\bar{x}_1 \wedge x_2 \wedge x_3)$
$(x_1 \wedge \bar{x}_2 \wedge x_3)$
$(x_1 \wedge x_2 \wedge \bar{x}_3)$
$(\bar{x}_1 \wedge \bar{x}_2 \wedge x_3)$
$(\bar{x}_1 \wedge x_2 \wedge \bar{x}_3)$
$(x_1 \wedge \bar{x}_2 \wedge \bar{x}_3)$
$(\bar{x}_1 \wedge \bar{x}_2 \wedge \bar{x}_3)$.

And 3-SAT (15/16) also have 8 possible clauses as same, but only 15/16 of clauses need to be true. If we can solve 3-SAT (15/16) in a black box, we only have to check if the rest of clauses are true, which is polynomial time. Therefore: 3-SAT (1) $\leq_p$ 3-SAT (15/16).

Therefore: 3-SAT (15/16) is NP-complete. Q.E.D.

Problem #2.

STEP 1: Show Dense Subgraph Problem is a NP-problem.

- Efficient certificate: A subset of vertex $V' \subset V$.
- Efficient certifier: Count the connected edges in V' and show the number of edges $\geq m$.
  Check vertex size $\leq k$.
  Check no repeating nodes.

STEP 2: Choose a known NP-complete problem: Clique problem.

STEP 3: In a clique problem, we are given an integer a, and we have to find a sub-graph that has at least a nodes and they're all adjacent to each other.

From the nature of both problems, we can convert: Clique problem's input to Dense subgraph problem's input:

$$\begin{cases} m = a. \\ k = a(a-1)/2. \end{cases}$$

If we can solve Dense Subgraph Problem, we can solve clique problem in $O(1)$ runtime. Therefore: Clique Problem $\leq_p$ Dense Subgraph Problem.

Therefore: Dense Subgraph Problem is NP-complete. Q.E.D.

Problem # 3.

Step 1: Prove SAT' is NP problem.

- Efficient certificate: A truth assignment is of polynomial length.
- Efficient certifier: We can check every clauses in polynomial time and count how many of them are true.

Step 2: Choose a known NP complete Problem: SAT problem.

STEP 3: Suppose the clauses for a SAT' problem is like this.

m clauses
$$\begin{cases} (x_1) \\ (x_1 \wedge x_2) \\ (\bar{x}_1 \wedge x_2 \wedge x_3) \\ \cdots \\ (x_1 \wedge x_2 \cdots \wedge \bar{x}_k \wedge \cdots \wedge x_n) \\ (\bar{x}_1 \wedge x_2 \cdots \wedge x_k \wedge \cdots \bar{x}_n) \end{cases}$$

these two can never be true

The last two clauses can not be true and there would be exactly m-2 clauses that are satisfied. And if we only take the first m-2 clauses and this can be a solution to a regular SAT problem.

Therefore, we can show that $SAT \leq_p SAT'$.

Therefore: SAT' is NP-complete. Q.E.D.

Problem # 4.

Step 1. Show this modified Vertex cover problem is NP.

- Efficient certificate: A vertex cover is of polynomial length
- Efficient certifier: Check all edges that only one node is in this Vertex cover.

  Cover size $\geq k$.

  No repeating nodes.

Step 2: Choose a known NP-complete problem: Vertex cover problem.

Step 3: Prove regular Vertex cover problem $\leq_p$ modified ~ problem.

Didn't figure this part out. ...

Therefore, this modified Vertex cover problem is NP complete.