CSCI 1010 - Fall'26 Theory of Computation Lorenzo De Stefani

HW9

November 19, 2026 at 11:59 pm

Reminder: Submit your assignment on Gradescope by the due date. Submissions must be typeset. Each page should include work for only one problem (i.e., make a new page/new pages for each problem). See the course syllabus for the late policy.

While collaboration is encouraged, please remember not to take away notes from any labs or collaboration sessions. Your work should be your own. Use of other third-party resources is strictly forbidden.

Please monitor Ed discussion, as we will post clarifications of questions there.

Problem 1

Given a graph $G = (V, E)$, we say that subset $V' \subseteq V$ is an "*even-degree k-clique*" in G if:

- $|V'| = k$,
- For all pairs $v_i, v_j \in V'$, $(v_i, v_j) \in E$ (i.e., V' is a k-clique).
- For all pairs $v_i, v_j \in V'$, v_i is connected to α vertices that are not connected to v_j (and vice versa).

Let $EDCLIQUE$ be the decision problem that, given as input instance an undirected graph G and a positive integer k, decides whether G includes an even-degree $k - clique$ or not.

Show that $EDCLIQUE$ is NP-complete.

Problem 2

Let *SATRange* be a language defined as follows:

$$SATRange = \{\langle \phi, lo, hi \rangle \mid hi \geq lo \geq 0, \\ \phi \text{ is a Boolean formula with } lo \leq k \leq hi \text{ satisfying assignments}, \\ lo \text{ and } hi \text{ are integers represented in binary}\}$$

1. Show that *SATRange* is NP-hard.
2. Show that *SATRange* is coNP-hard.
3. Show that if *SATRange* is in NP, then NP=coNP.
4. Give a polynomial-time algorithm that, given a Boolean formula ϕ as input, determines the number of satisfying assignments for ϕ using an oracle for the language *SATRange*.
   Note: an oracle for a language is essentially a black box capable of instantly deciding whether or not a string is in that given language.

Problem 3

A 2-vertex coloring is a function that assigns the colors *blue* and *red* to the vertices of a graph. A 2-vertex coloring is called *proper* if no two vertices that are connected by an edge receive the same color. Let

$$\text{2COLOR} = \{G \mid G \text{ has a proper 2-vertex coloring}\}$$

We now give a flawed "proof" that 2COLOR is NP-hard. Please pinpoint the flaw in the argument and discuss where it logically goes wrong. Also: while this proof is wrong, do you suspect there might be another correct proof that 2COLOR is indeed NP-hard? Why or why not?

Here's the false proof: The reduction is from 3SAT.

We construct the following gadget: Given a formula ψ, we construct the graph G_ψ as follows: G_ψ has a central vertex v. Then for each clause c_i in ψ, we join a new vertex c_i to v, and connect new vertices a_i and b_i each to c_i. If there is no satisfying assignment for ψ, we additionally add edges from each a_i to its corresponding b_i. Then, if ψ is satisfiable, we can 2-color G_ψ by assigning the color *red* to v, and all the vertices labelled a_i and b_i, while assigning the color *blue* to all the vertices labeled c_i. On the other hand, if ψ is unsatisfiable, then the vertices a_i, b_i, c_i form a triangle and the graph is not properly 2-colorable. Thus G_ψ is 2-colorable iff ψ was satisfiable, so we have reduced 3 SAT to 2COLOR, and since 3SAT is NP-hard, so is 2COLOR.