HW7 - Complexity Theory and P vs. NP

COMP361 — Suhas Arehalli

Spring 2025

Instructions

Note that unlike other assignments, there are **no graded problems** on this assignment. Think of it as a set of practice problems to guide your preparation for the next exam. These are organized by topic, covering the relevant topics we haven't seen a HW problem on.

To prove that...

- ...a language $L \in P$, provide a polynomial time TM that decides L
- ...a language $L \in NP$, provide a polynomial-time verifier or polynomial-time Nondeterministic Turing Machine (NTM).
- ...a language L is NP-COMPLETE, show $L \in NP$ and provide a reduction from $SAT/3SAT$ (or other known NP-COMPLETE problem) to L.

Questions

1. *(Sipser 7.9)* Consider

$$TRIANGLE = \{\langle G \rangle \mid G \text{ contains a 3-CLIQUE } \}$$

   Show that $TRIANGLE \in P$.

2. *(Sipser 7.23)* Consider a CNF formula ϕ with m variables and c clauses. Show that you can construct, in polynomial time, an NFA with $O(cm)$ states that accepts all non-satisfying assignments to ϕ, represented as a boolean string of length m. Conclude that $P \neq NP$ implies that NFAs cannot be minimized in polynomial time.

3. *(Sipser 7.34)* Let

$$U = \{\langle M, x, \#^t \rangle \mid \text{NTM } M \text{ accepts } x \text{ within } t \text{ steps on some branch}\}$$

   Show that U is NP-Complete.

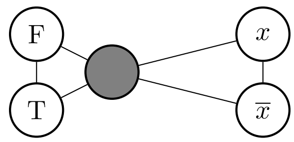

Figure 1: A gadget that enforcing an assignment of colors for true and false to a variable x and it's complement $\overline{x}$. Note that Sipser calls the triangle/3-clique on the left a *palette* to indicate that it captures our 3 distinct colors: T, F, and ■

.

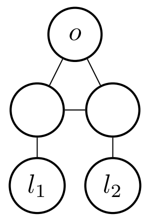

Figure 2: A gadget that only allows vertex o to be colored true iff at least one of l_1 and l_2 is colored true.

4. A k-coloring of a graph is an assignment of one of k colors to each vertex such that no edge connects two vertices of the same color. Consider

$$3COLOR = \{\langle G \rangle \mid G \text{ is 3-colorable}\}.$$

   Prove that $3COLOR$ is NP-COMPLETE via a reduction. To do this, consider the gadgets in Fig. 1 and 2 and piece them together to construct a graph that is 3-colorable iff a boolean formula is satisfiable.

5. *(Sipser 7.45)* Show that if $P = NP$, then every language $A \in P$, other than $A = \emptyset$ or $A = \Sigma^*$, is NP-COMPLETE.