CS46 practice problems 10

These practice problems are an opportunity for discussion and trying many different solutions. It is **not counted towards your grade**, and **you do not have to submit your solutions.** The purpose of these problems is to get more comfortable with reasoning and writing about P, NP, and polynomial-time reductions.

If you are stumped or looking for guidance, **ask**.

1. A **vertex cover** in a graph $G = (V, E)$ is a subset $S \subset V$of vertices where every edge of G has at least one endpoint in the subset.

   $$\textsc{VertexCover} = \{\langle G, k\rangle \mid G \text{ has a } k\text{-node vertex cover}\}$$

   An **independent set** in a graph G is a subset of vertices with no edges between them.

   $$\textsc{IndependentSet} = \{\langle G, k\rangle \mid G \text{ contains an independent set of } k \text{ vertices}\}$$

   Show that $\textsc{IndependentSet} \leq_p \textsc{VertexCover}$.

2. A Boolean formula is in **conjunctive normal form** (CNF) if it is written as the conjunction of clauses, for example:

   $$(x_1 \vee x_2) \wedge (\overline{x}_2 \vee \overline{x}_3 \vee x_4) \wedge (x_5 \vee \overline{x}_1 \vee x_6) \wedge (x_3)$$

   (Recall that a **literal** is a Boolean variable or a negated Boolean variable, like x or $\overline{x}$, and a **clause** is a disjunction of literals, like $x \vee y \vee \overline{z}$. The symbol "$\vee$" means "or"; the symbol "$\wedge$" means "and".) A formula is **satisfiable** if there is a truth assignment (giving a truth value to each variable) which makes the entire formula evaluate to TRUE.

   (a) Define the language:

   $$L = \{\langle \phi\rangle \mid \phi \text{ is a satisfiable CNF formula where each variable appears at most twice}\}$$

   Show that $L \in P$.

   (b) For a CNF formula ϕ with m variables and c clauses, show you can construct in polynomial time an NFA with $O(cm)$ states that accepts all *non*satisfying assignments, represented as binary strings of length m.
   (This implies that if $P \neq NP$, then NFAs cannot be minimized in polynomial time.)

3. A **coloring** of a graph is an assignment of colors to its nodes so that no two adjacent nodes are assigned the same color.

   Define 3-COLOR as:

   $$3 - \textsc{Color} = \{\langle G\rangle \mid G \text{ is colorable with three colors}\}$$

   Show that 3-COLOR is NP-complete. (Hint: reduce from 3-SAT and use the subgraphs given in the textbook hint, page 325. Overall, your construction should preserve the property that: there exists a valid 3-coloring if and only if there is a satisfying assignment.)