COSC 240, Fall 2017: Problem Set $\#7$

Assigned: Wednesday, 11/29.
Due: Wednesday 12/6, at the beginning of class (hand in hard copy).
Lectures Covered: 22 to 25
Academic Integrity: You can work with other people in the class but you must write up your own answers in your own words. You can also use the textbook and talk to the professor. You may not use any other resources (e.g., material found online) or talk to people outside the class about these problems. See the syllabus for details on the academic integrity policy for problem sets.

Problems

1. In class we proved that there does not exist an algorithm that *decides* the following language:

$$A_{ALG} = \{\langle A, w\rangle \mid A \text{ is an algorithm that accepts input } w\}.$$

   We did not, however, say anything about whether or not there exists an algorithm that *accepts* it (a weaker notion). To complete this discussion, describe an algorithm that *accepts* language A_{ALG}.

   (A high-level description is fine. There is no need to use pseudocode).

2. Let $\overline{A_{ALG}}$ be the language that contains every string that is *not* in A_{ALG}. In class we proved A_{ALG} cannot be decided. In the above problem you proved that A_{ALG} can be accepted. Use these two facts to prove that $\overline{A_{ALG}}$ cannot be accepted.

   (*Hint:* Assume you are allowed to run two subroutines in parallel.)

3. Prove that each of the following three languages are in NP:

   (a) $COMPOSITES = \{\langle x\rangle \mid x = pq, \text{ for integers } p, q > 1\}$.
   (b) $NEQ = \{\langle \phi, \phi'\rangle \mid \phi$ and ϕ' are boolean formulas that are *not* equivalent$\}$.
   (c) $BADPAIRS = \{\langle S, k\rangle \mid S$ is a set containing natural numbers, $k \in S$, for every $x, y \in S, xy \neq k\}$.

4. Prove that A_{ALG} is not in NP.

   (*Hint #1:* Start by assuming for contradiction that A_{ALG} *is* in NP. Then build on this assumption to produce a contradiction to a result we previously proved about this language.)

   (*Hint #2:* Recall that our definition of a polynomial-time verifier requires that there is some polynomial $f(n)$, such that when the verifier is run on any input $\langle x, y\rangle$, where $n = |x|$, it terminates in no more than $f(n)$ steps. You can assume, without loss of generality, that $|\langle x, y\rangle| \leq f(n)$.)

5. Prove that the language $5SAT = \{\langle \phi\rangle \mid \phi$ is a boolean formula in 5-cnf with a satisfying assignment$\}$ is NP-complete.

   (*Note:* 5-cnf is defined the same as 3-cnf except there are 5 literals in each clause instead of 3.)