Problem Set 6

Read Sections 9.1, 9.2, 10.2, 10.4 through Theorem 10.33.

Skip the section on Primality (Theorem 10.6 thru Theorem 10.9) and the proof of Lemma 10.30.

1. Let \( EQ_{BP} = \{ \langle B_1, B_2 \rangle \mid B_1 \text{ and } B_2 \text{ are equivalent branching programs} \} \).
   Show that \( EQ_{BP} \) is coNP-complete.

2. Define a two-headed finite automaton (2DFA) to be a deterministic finite automaton that has two read-only, bidirectional heads that start at the left-hand end of the input tape and can be independently controlled to move in either direction. The tape of a 2DFA is finite and is just large enough to contain the input plus two additional blank tape cells, one on the left-hand end and one on the right-hand end, that serve as delimiters. A 2DFA accepts its input by entering a special accept state. For example, a 2DFA can recognize the language \( \{ a^n b^n c^n \mid n \geq 0 \} \). Prove that P contains a language that is not recognizable by a 2DFA.

3. Let \( SAT2 = \{ \langle \phi \rangle \mid \phi \text{ is a Boolean formula that has exactly two satisfying assignments} \} \).
   Show that \( SAT2 \in P^{SAT} \). (Hint: Show that \( SAT2 = A \cap B \) where \( A \in NP \) and \( B \in coNP \).)

4. This problem considers deterministic, polynomial-time oracle Turing machines that have access to an oracle for SAT.
   (a) Describe such a SAT-oracle Turing machine that takes as input an undirected graph \( H \), and outputs the size (number of nodes) of a largest clique in \( H \).
   (b) Describe another such SAT-oracle Turing machine that takes as input an undirected graph \( H \), and outputs the list of the nodes in one of the cliques in \( H \) of the largest size.

5. If \( B \) is a branching program on variables \( x_1, \ldots, x_m \) we can write an input assignment to the variables as a binary string \( w \) of length \( m \). Let \( EQ_{ROBP, DFA} = \{ \langle B, A \rangle \mid B \text{ is a read-once branching program on } m \text{ inputs and } A \text{ is a DFA, where for all } w \text{ of length } m, B(w) = 1 \text{ iff } A \text{ accepts } w \} \). Show that \( EQ_{ROBP, DFA} \in BPP \).

6. The class RP is a subset of BPP, where the probabilistic polynomial time decider never accepts for inputs outside the language, thereby exhibiting one-sided error. More formally, RP is the collection of languages \( A \) for which a probabilistic polynomial time decider, accepts with probability at least \( \frac{2}{3} \) (or equivalently \( \frac{1}{2} \)) for inputs in \( A \) and accepts with probability 0 for inputs not in \( A \). For example, our proof that \( EQ_{ROBP} \in BPP \) actually shows that \( EQ_{ROBP} \in coRP \). Prove that if \( NP \subseteq BPP \) then \( NP = RP \).
   (Hint: An RP machine should accept only when it is certain that its input is in the language. How can we be certain that a formula \( \phi \) is satisfiable?)

7. (Optional) Give an example of a language \( A \in L \) where \( A^* \) is NL-complete.
   (Hint: Leveled graphs, defined in the book’s solution to Problem 8.34, may be helpful. Careful, this problem is very difficult, though the solution is short.)

The Final Exam will be held Tuesday, December 17, 2015, 9am – Noon, Johnson Track.
It covers Chapters 1, 2 (except 2.4), 3, 4, 5, 6.1, 7, 8, 9.1, 9.2, 10.2 (except the part on Primality),
and 10.4 through Theorem 10.33.