18.404/6.840 Michael Sipser
Theory of Computation
Due via Gradescope 2:30pm sharp, Thursday, December 3, 2020 **Problem Set 6**

Read Sections 9.1, 9.2, 10.2, 10.4 through Theoreom 10.33.
Skip the section on Primality (Theorem 10.6 thru Theorem 10.9) and the proof of Lemma 10.30.

1. Let $EQ_{\mathsf{BP}} = \{\langle B_1, B_2\rangle |\ B_1$ and B_2 are equivalent branching programs$\}$.
   Show that EQ_{BP} is coNP-complete.

2. Let $SAT2 = \{\langle\phi\rangle |\ \phi$ is a Boolean formula that has exactly two satisfying assignments$\}$.
   Show that $SAT2 \in \mathrm{P}^{SAT}$.

3. Describe a deterministic, polynomial-time SAT-oracle Turing machine M^{SAT} that takes as input a directed graph G and nodes s and t, and outputs a Hamiltonian path from s to t if one exists. If none exist, then M^{SAT} outputs **No Hamiltonian path**.

4. Let ICFL be the class of languages that can be expressed as the intersection of two context free languages. In other words $\mathsf{ICFL} = \{A |\ A = B \cap C$ for some CFLs B and $C\}$.

   (a) Prove $\mathsf{ICFL} \subseteq \mathrm{P}$.

   (b) Prove that P contains some language which is not in ICFL.
   (Hint: a theorem we proved in lecture is useful here.)

5. Let $EQ_{\mathsf{NFA}} = \{\langle A, B\rangle |\ A$ and B are NFAs and $L(A) = L(B)\}$.
   Show that EQ_{NFA} is PSPACE-complete.

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 $EQ_{\mathsf{ROBP}} \in \mathrm{BPP}$ actually shows that $EQ_{\mathsf{ROBP}} \in \mathrm{coRP}$. Prove that if $\mathrm{NP} \subseteq \mathrm{BPP}$ then $\mathrm{NP} = \mathrm{RP}$.
   (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 ϕ is satisfiable?)

7.* (Optional) Suppose that A and B are two oracles. One of them is an oracle for $TQBF$, but you don't know which. Give an algorithm that has access to both A and B, and that is guaranteed to solve $TQBF$ in polynomial time.

Final exam: Thursday, December 17, 2020, 3 hours, start time flexible.
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 Theoreom 10.33.