Aim to do 5 problems. You may also choose up to two problems from previous assignments (provided that you did not do them already).

1. Let $H$ be the language consisting of tuples $< x, 1^{2^n}, 1^s >$ so that $x$ is a boolean string of length less than $2^n$ that is the prefix of some truth table of length $2^n$ whose corresponding boolean function cannot be computed by any circuit of size $s$. Show that $H$ is in $\Sigma_2^P = NP$. (Very hard to parse, easy to do.)

2. Show that a graph is bipartite iff it has no odd length cycle. Use this to show that bipartite testing is in NL.

3. Show that the conversion of a number represented in balanced $p$-ary notation to binary notation is in $NC^1$.

4. A strong nondeterministic TM is one that has three possible halt states: "yes", "no", or "maybe". We say such a machine decides $L$ in polynomial time if all computations run in polynomial time, and if the following holds: if $x \in L$, then all computations end up with "yes" or "maybe", and at least one ends up "yes". If $x \notin L$, then all computations end up in "no" or "maybe", but at least one ends up "no". Prove that $L$ is decided by a strong nondeterministic TM in polynomial time iff $L \in NP \cap coNP$.

5. In fact, we say that $A$ is strong-nondeterministically reducible to $B$, $A \leq_{SN}^B$, if $A \in NP^B \cap coNP^B$. Show that the set of $\leq_{SN}$-complete sets for NP is precisely $HP^1$. 

6. A pattern is a string over the alphabet $\{0, 1, *\}$. A pattern $\pi$ covers a string $w$ if $w$ can be obtained from $\pi$ by replacing each occurrence of * in $\pi$ with either a 0 or 1. (For example, 01** covers 0100, 0101, 0110, and 0111.) Define the problem PATTERN as follows: given a set $\Pi$ of patterns, each of length $n$, determine if there exists a string $s$ of length $n$ such that no pattern in $\Pi$ covers $s$. Show that PATTERN is NP-complete.

7. Let $A$ be the set of properly nested parentheses. For example, ((())) is in $A$ while ())( is not. Show that $A$ in in $L$ (log-space).

8. Show that if every NP-hard problem is also PSPACE-hard, then NP=PSPACE. (This is pretty easy.)

9. optional: Prove that $E = NC^2$. (This one might be hard.)

Comments:

- Recall from assignment 3 that $H_1^P = \{ A | \Sigma_2^P \subseteq NP^A \}$.
- $\leq_{SN}$ forms a type of nondeterministic Turing reduction. A many–one analogue can be seen in the $\leq^\gamma$ reductions mentioned in exercise 10.4.2, pp 235-236. You may replace the fourth problem above, with this one.