COS 487: Theory of Computation	**Fall 2008**
Assignment #5	
Due: Tuesday, December 2	*Sanjeev Arora*

(Contains 8 questions)

1. Consider the language PUZZLE defined in Prob. 7.26 (of both the old and new editions of the book). Now consider a two player version – each player starts with an ordered stack of puzzle cards. The players take turns placing the cards in order in the box and may choose which side faces up. Player I wins if, in the final stack, all hole positions are blocked, and Player II wins if some hole position remains unblocked. Show that the problem of determining which player has a winning strategy for a given starting configuration of the cards is PSPACE-complete.

2. Show that A_{NFA} (the problem of deciding if an NFA accepts a given string) is NL-complete.

3. Let *UPATH* be the analogue of *PATH* for undirected graphs. Let *BIPARTITE* be the problem of deciding if a graph is bipartite (i.e., it can be colored using two colors such that no two adjacent vertices get the same color).

Show that $BIPARTITE \leq_L UPATH$. Recall that in the precept it was mentioned that *UPATH* is in L, consequently *BIPARTITE* is in L.

4. Show that if $\mathsf{NP} = \mathsf{P}^{SAT}$, then $\mathsf{NP} = \mathsf{coNP}$.

5. Consider the function pad : $\Sigma^* \times \mathcal{N} \to \Sigma^* \#^*$ that is defined as follows. Let $\text{pad}(s, l) := s\#^j$, where $j = \max(0, l - m)$ and m is the length of s. Thus $\text{pad}(s, l)$ simply adds enough copies of the new symbol $\#$ to the end of s so that the length of the result is at least l. For any language A and function $f : \mathcal{N} \to \mathcal{N}$, define the language $\text{pad}(A, f(m))$ as

$$\text{pad}(A, f(m)) := \{\text{pad}(s, f(m)) \ : \ \text{where } s \in A \text{ and } m \text{ is the length of } s\}$$

(a) (Warmup) Prove that if $A \in \textsc{Time}(n^6)$ then $\text{pad}(A, n^2) \in \textsc{Time}(n^3)$.

(b) Prove that, if $\mathsf{NEXPTIME} \neq \mathsf{EXPTIME}$, then $\mathsf{P} \neq \mathsf{NP}$. *Hint(surprise!):* Use padding.

6. Define the ***unique-sat*** problem to be

$$USAT := \{\phi \ : \ \phi \text{ is a boolean formula that has a single satisfying assignment}\}$$

Show that $USAT \in \mathsf{P}^{SAT}$.

7. 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.

8. Let HALT denote the halting problem. Show that $\mathsf{P}^{\text{HALT}} = \mathsf{NP}^{\text{HALT}}$. (i.e., a deterministic machine is as powerful as a non-deterministic one if they both have access to an oracle for the Halting problem).