CSE 531: Complexity theory | Winter 2007
Problem Set #3 | Instructor: Venkatesan Guruswami
Due in class on Wednesday, **February 14, 2007**,
or to Prasad by 5pm on Friday, Feb 16.

Instructions: Same as for Problem set 1.

Each problem is worth 10 points unless noted otherwise.

1. Following the characterization of NP as problems whose solutions can be verified in P with the help of a certificate, we can imagine that perhaps NL can be characterized as the class of logspace verifiable languages defined as follows. Define a language A to be *logspace verifiable* if there exists a relation $R \subseteq \Sigma^* \times \Sigma^*$ and an integer k such that

   - $x \in A \iff \exists y$ s.t. $\langle x, y \rangle \in R$
   - $\langle x, y \rangle \in R \implies |y| \leq |x|^k$, and
   - $R \in \mathsf{LOGSPACE}$.

   However, this does not (or rather is unlikely to) characterize NL.

   (a) Prove that, in fact, the class of logspace verifiable languages is exactly NP.

   (b) Suppose we now restrict the verifier to have only left-to-right read-once access to the certificate. In other words, the verifier is given x on the read-only input tape and the certificate y on a separate read-only tape in which the head can never move left. In addition, it has a read/write tape with $O(\log |x|)$ cells.

   Prove that if a language A has such a restricted logspace verifier if and only if $A \in \mathsf{NL}$.

   (c) **(No need to turn this part in)** Give an alternate description of the $\mathsf{NL} = \mathsf{coNL}$ proof from class using a restricted logspace verifier (as defined in Part 1b) that can verify the non-existence of an s-t path in a directed graph.

2. Prove that $\mathsf{NP} \neq \mathsf{SPACE}(n)$.

   Hint: Assume that $\mathsf{NP} = \mathsf{SPACE}(n)$ and get a contradiction to the space hierarchy theorem.

3. Prove that $\mathsf{NTIME}(n) \neq \mathsf{NTIME}(n^2)$.

   Hint: The idea is to use diagonalization, except it is not clear how a nondeterministic machine running in time $O(n^2)$ can flip the answer of a nondeterministic machine running in $O(n)$ time (since it may have to examine $2^{\Omega(n)}$ possible sequences of nondeterministic choices). Instead use a "sloppier" diagonalization to build an NDTM N that differs from the i'th NDTM M_i on input 1^m for *some* m in the range $f(i) < m \leq f(i+1)$. The key is to pick a very rapidly growing integer-valued function f, so that the machine N, on input $1^{f(i+1)}$ has enough time to do the opposite of what M_i does on input $1^{f(i)+1}$.

4. Prove that there exists an oracle C for which $\mathsf{NP}^C \neq \mathsf{coNP}^C$.

5. The cat-and-mouse game is played by two players, "Cat" and "Mouse," on an arbitrary undirected graph. At a given point each player occupies a node of the graph. The players take turns moving to a node adjacent to the one that they currently occupy. A special node of the graph is called "Hole". Cat wins if the two players ever occupy the same node. Mouse

wins if it reaches the Hole before the preceding happens. The game is a draw if the two players ever simultaneously reach positions they previously occupied. Let

$$HAPPYCAT = \{\langle G, c, m, h \rangle \mid G, c, m, h, \text{ are respectively a graph, and positions of the Cat, Mouse, and Hole, such that Cat has a winning strategy, if Cat moves first}\}\ .$$

Prove that $HAPPYCAT$ is in P.

6. Prove that if $\mathsf{PH} = \mathsf{PSPACE}$, then the polynomial time hierarchy has only finitely many distinct levels, i.e., $\mathsf{PH} = \Sigma_k^P$ for some $k \geq 1$.