Due: Thursday September 15, 2005

The purpose of this homework set is to have you think about the oracle access mechanism of deterministic and nondeterministic Turing machines (TM). Although we use TMs as a formal model, you should nevertheless describe your solutions at a high-level — usually in paragraph form, use pseudocode only if really necessary. In particular, you do not need to describe the head movements of a TM. However, since we are particularly concerned about oracle queries for this homework set, be fairly explicit what queries are asked and how the queries are constructed.

1. Characterization of $\mathrm{NP}^{\mathrm{NP}}$ languages.

We want to show that for every language $L \in \mathrm{NP}^{\mathrm{NP}}$, there exists polynomials $p()$ and $q()$ and a polynomial-time predicate $R()$ such that

$$x \in L \iff \exists y \in \Sigma^{p(|x|)},\ \forall z \in \Sigma^{q(|x|)},\ R(x, y, z). \tag{1}$$

Without loss of generality, let L be recognized by an NP machine N using a SAT oracle.

a. Consider an accepting computation path of $N^{\mathrm{SAT}}(x)$. The NP machine N makes some nondeterministic moves and makes some oracle queries along this path. Construct an equivalent NP machine N' that makes all the nondeterministic moves before making any oracle queries. (*Hint:* N' should guess the replies from the SAT oracle and verify its guesses later.)

b. The machine you constructed above might make many oracle queries along each computation path. Construct an equivalent NP machine N'' that makes only 2 oracle queries to SAT on each computation path.

c. Construct an equivalent NP machine N''' that makes only 1 oracle query SAT on each computation path.

d. Argue that N''' always accepts if and only if the reply from the SAT is "no".

e. Show how to construct the polynomial-time predicate $R()$ and argue that Equation (1) holds.

2. Sparse Sets versus Polynomial Advice

A set S is *sparse* if there exists a polynomial $p()$ such that for all n, the number strings in S of length n is bounded by $p(n)$. We want to show that

$$L \in \mathrm{P/poly} \iff \exists S,\ S \text{ is a sparse set and } L \in P^S.$$

a. Suppose that $L = L(M^S)$ for some polynomial-time TM M with running time $r(n)$. For a fixed length n, what portion of the oracle S does M require on inputs of length n? If you encode this information in an advice string, how long is the advice? Argue that $L \in \mathrm{P/poly}$.

b. Suppose that $L \in \mathrm{P/poly}$ via a polynomial-time TM M_1 and polynomial advice function f. Construct a sparse set S and a polynomial-time TM M_2 such that $L = L(M_2^S)$. *N.B.:* Recall that the oracle S can only answer the yes/no question regarding the membership of strings in S.