6.045J/18.400J: Automata, Computability and Complexity	Prof. Nancy Lynch
Homework 3	
Due: March 2, 2005	*Vinod Vaikuntanathan*

This problem set contains some harder-than-usual problems. In solving them, you can call upon everything you have learned so far about finite-state automata and regular languages.

Problem 1: **Distinguishable strings and indices** (From Sipser Problems 1.51 and 1.52) Let x and y be strings and let L be *any* language (not necessarily regular). We say that x and y are *distinguishable by* L if some string z exists such that exactly one of the strings xz and yz is in L. On the other hand, if for all strings z, xz is in L if and only if yz is in L, we say that x and y are *indistinguishable* by L. If x and y are indistinguishable by L, we write $x \equiv_L y$.

(a) Show that $\equiv_L$ is an equivalence relation.

Now let L be a language and X a set of strings. We say that X is *pairwise distinguishable* by L if every two distinct strings in X are distinguishable by L. Define the *index* of L to be the maximum number of elements in any set that is pairwise distinguishable by L. In other words, the index of L is equal to the number of equivalence classes in L, which may be finite or infinite.

(b) Let L_1 be the regular language $(001)^*00$. What is the index of L_1? Describe the equivalence classes.

(c) Build a DFA for L_1 with states corresponding to the equivalence classes (i.e., the number is states is equal to the index of L_1).

(d) Let L_2 be the non-regular language $\{0^n1^n : n \geq 1\}$. What is the index of L_2? Describe the equivalence classes.

(e) Now consider an arbitrary language L. Prove that if L is recognized by a DFA with k states, then L has index at most k.

(f) Again consider an arbitrary language L. For L with index k, describe how to construct a DFA with k states.

We can conclude from this problem that a language L is regular if and only if it has a finite index. Moreover, its index is the size of the smallest DFA recognizing it.

Problem 2: **Inequivalent DFAs** Suppose that two DFAs $M_1 = (Q_1, \{0,1\}, \delta_1, q_01, F_1)$ and $M_2 = (Q_2, \{0,1\}, \delta_2, q_02, F_2)$ over alphabet $\{0,1\}$ recognize different languages.

(a) In terms of the sizes of the state sets Q_1 and Q_2, determine an upper bound u on the length of the smallest string on which machines M_1 and M_2 must give different answers (accept vs. reject). That is, determine some u such that M_1 and M_2 must actually give different results for some string of length $\leq u$.

(b) Prove your answer to (a).