Problems 1–4. Due Tuesday, April 21, 1998

Problem 1.

A. A **Moore machine** is like a DFA except that each state is annotated by character. Unlike a DFA, which computes a predicate on strings, a Moore machine computes a string-valued function on strings. Formally define Moore machines and the function computed by a Moore machine.

B. A **Mealey machine** is like a DFA except that transition out of a state annotated by character. As with a Moore machine, a Mealey machine computes a function on strings. Formally define Mealey machines and the function computed by a Mealey machine.

C. Show that a function f is computable by a Moore machine iff f is computable by a Mealey machine.

Problem 2.

Let $L \subseteq \Sigma^*$ be a language and define the following equivalence relation on Σ^*: $x \equiv_L y$ if for all $s \in \Sigma^*$, $xs \in L \Leftrightarrow ys \in L$. The number of equivalence classes of $\equiv_L$ is called the *index* of L. This could be a number, or it could be infinity.

A. Show that L is regular if and only if L has finite index.

B. Let $L = \{x \in \{0,1\}^* : x \text{ contains at least two 0's but not two consecutive 0's}\}$. What is the index of L?

Problem 3. Using Problem 2, find a proof different from the one given in class of the following: L is regular iff L is decidable by a 2-Way Finite Automaton.

Problem 4. Give a family of languages E_n such that E_n can be recognized by an n-state NFA but requires at least c^n states on a DFA for some $c > 1$. Prove that your languages have this property.