6.045J/18.400J: Automata, Computability and Complexity Prof. Nancy Lynch, Nati Srebro

Recitation 4: Algorithms for Automata

February 26, 2003 *Susan Hohenberger*

Problem 1: **Quiz Questions?** Quiz Statistics ...

Problem 2: **Equivalence Classes.** 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. In the opposite case, 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 $x \equiv_L y$.

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.

Let's consider the language L: $(001 \cup 110)^*$. What is the index of L?
Describe the equivalence classes. Can we build a DFA for L with *less* states than the index of L?

Problem 3: **Procedures for testing properties of FAs**

1. E_{DFA} - We saw in class (and it is on page 154 of Sipser) how to test if a DFA M recognizes exactly the empty language. Now, let's think about how to test if $L(M) = \emptyset$ by running M on all strings up to a certain size. What size works?

2. F_{DFA} - We saw in class (but it is not in Sipser) how to test if a DFA M recognizes a finite language by seeing if it rejects all strings of length $|M|$ to $2|M|$. Now, let's think about how to test if M recognizes a finite language via a graph-based search. What are we looking for?