Problem Set 2 — Due April 15, 2004

Problem 1. Give DFAs for the following languages. Assume the alphabet for the DFA is \( \Sigma = \{0, 1\} \).

(a) The set of all strings with 010 as a substring.
(b) The set of all strings which do not have 010 as a substring.
(c) The set of all strings which have an even number of 0's or an even number of 1's.
(d) The complement of \( \{1, 10\}^* \).
(e) The binary encodings of numbers divisible by 3: \( \{0\}^* \circ \{\varepsilon, 11, 110, 1001, 1100, 1111, \ldots \} \).

Problem 2. State whether the following proposition are true or false, proving each answer.

**Part A.** Every DFA-acceptable language can be accepted by a DFA with an even number of states.

**Part B.** Every DFA-acceptable language can be accepted by a DFA whose start state is never visited twice.

**Part C.** Every DFA-acceptable language can be accepted by a DFA no state of which is ever visited more than once.

**Part D.** Every DFA-acceptable language can be accepted by a DFA with only a single final state.

Problem 3. The following problem helps explain our emphasis on looking at languages instead of more general computational problem. It gives a tiny bit of evidence in support of the thesis that studying “the ability to decide” is essentially the same as studying “the ability to compute”.

Suppose you are given an oracle (a “black box” or “magic subroutine”) that decides in unit time if a given graph \( G = (V, E) \) is three-colorable. (We defined when a graph is three-colorable in lecture 1.) Show how to use the algorithm to find a three-coloring of \( G \), when one exists, in \( O(|V|) \) time.

Problem 4. Suppose that \( L \) is DFA-acceptable. Show that the following languages are DFA acceptable, too.

**Part A.** \( \text{Max}(L) = \{ x \in L : \text{there does not exist a } y \in \Sigma^+ \text{ for which } xy \in L \} \).

**Part B.** \( \text{Echo}(L) = \{ a_1a_2a_2 \cdots a_na_n \in \Sigma^* : a_1a_2 \cdots a_n \in L \} \).