Introduction to Theory of Computing	Problem Set 1
CS 481 Fall 2004	Due Wednesday, September 8, 2004

For each problem set please write both your name and your cornell.edu email address on top.

For some problems you will need to construct finite automata. Provide the required formal definitions, and also a brief English explanation of the main idea (e.g., what the states represent, etc). You may defines the states and transition function by drawing a transition diagram.

(1) Let $M = (Q, \Sigma, \delta, s, F)$ be an arbitrary DFA. Prove that for all $x, y \in \Sigma^*$ and $q \in Q$, $\hat{\delta}(q, xy) = \hat{\delta}(\hat{\delta}(q, x), y)$, where $\hat{\delta}$ is the extended version of δ defined on all strings as described in class. (*Hint.* Induction on $|y|$.)

(2) Design deterministic finite automata for each of the following sets:

(a) the set of strings in $\{0, 1\}^*$ not containing the substring 001;

(b) the set of strings in $\{a\}^*$ whose length is divisible by either 2 or 3;

(c) the set of strings $x \in \{0, 1\}^*$ such that $\#0(x) \cdot \#1(x)$ is even;

(d) For this part we will think of strings in $\{0, 1\}^*$ as representing numbers in *backwards binary.* Backwards binary the same as the traditional binary encoding, expect that we will read the digits backwards. So the leftmost position corresponds to the 1's place, the next is the 2's place, the 4's place, etc. So 011 is 6 in backwards binary (because 110 is 6 in the regular binary encoding). In general the sequence $\sigma_1\sigma_2, \ldots, \sigma_n$ represents the number $\sum_{i=1}^{n} \sigma_i 2^{i-1}$.

Design deterministic finite automata for the set of strings that represent numbers divisible by 3 in the backwards binary representation.

(3) Show that for any regular language $L \subseteq \Sigma^*$ the language Padded(L) is also regular, where Padded(L) is defined as

$$\text{Padded}(L) = \{\sigma_1\sigma_2\sigma_3\sigma_4 \ldots \sigma_{2n} | \sigma_i \in \Sigma, \text{ and } \sigma_2\sigma_4 \ldots \sigma_{2\text{n}} \in \text{L}\}$$

(4) Note that if we set $A = \emptyset$, and $B = \{\varepsilon\}$ then $A^* = B^*$. Are there any non-empty disjoint languages A and B such that $A^* = B^*$? Prove your answer.