CSE 105: Automata and Computability Theory **Autumn 2013**

Problem Set #1

Due: Thursday, October 10th, 2013

Problem 1 Warmup.

- **a.** Prove that if a language L contains finitely many elements then it is regular.
- **b.** If a language L contains infinitely many elements, can it still be regular?
- **c.** Give the state diagram for a DFA accepting the language consisting of all strings over $\Sigma = \{0, 1\}$ that do not end with the string "0101". For example, the string "010" is in the language, whereas "000101" is not.

Problem 2 We defined DFAs to have a set F of accepting states. In this problem we consider a variant of DFAs that have exactly one accepting state, q_{accept}. We call this variant of DFAs *one-DFAs*.

- **a.** Give a formal definition of one-DFAs, including the parts of a one-DFA, what it means for a one-DFA to accept a string w, and the language of a one-DFA.
- **b.** Show that for any one-DFA M_1 there exists an equivalent DFA M (i.e., one such that $L(M_1) = L(M)$).
- **c.** Show that the converse is not true: there exist regular languages (i.e., languages recognized by some DFA) that are not recognized by *any* one-DFA.
- **d.** Consider an analogous definition of *one-NFAs*, NFAs with exactly one accepting state. How do one-NFAs compare to NFAs?

Problem 3 We say that a language is *2-regular* if it is recognized by some DFA that has at most 2 states.

- **a.** Show that the class of 2-regular languages is closed under complement.
- **b.** Show that the class of 2-regular languages is *not* closed under union.
- **c.** Show that the class of 2-regular languages is *not* closed under intersection.

Hint: For parts b. and c., Try enumerating all 2-regular languages over a unary alphabet. De Morgan's laws may also save you some work.