Problem Set 2 – Due Friday, April 12, 2013

Problem 1 Draw a smallest DFA M_1 for $L_1 = \{w \in \{a,b\}^*\colon |w| \equiv 1 \pmod 3\}$. Draw a smallest DFA M_2 for $L_2 = \{w \in \{a,b\}^*\colon w \text{ ends in `}aa\text{'}\}$. Applying the product construction, draw the machine M whose language is $L_1 \cup L_2$. To make your method clear, name the states of M_1, M_2, and M. What would change if you wanted a machine for $L_1 \cap L_2$?

Problem 2 Given a mapping $\delta : Q \times \Sigma \to Q$, we defined $\delta^* : Q \times \Sigma^* \to Q$ by asserting that $\delta^*(q, \varepsilon) = q$ and $\delta^*(q, ax) = \delta^*(\delta(q,a), x)$. Prove that $\delta^*(q, xy) = \delta^*(\delta^*(q,x), y)$ for all $q \in Q$ and $x, y \in \Sigma^*$. Carefully justify each step.

Problem 3 State whether the following propositions are true or false, proving each answer.

(a) Every DFA-acceptable language can be accepted by a DFA with an even number of states.

(b) Every DFA-acceptable language can be accepted by a DFA whose start state is never visited twice.

(c) Every DFA-acceptable language can be accepted by a DFA no state of which is ever visited more than once.

(d) The language $L = \{x \in \{a,b\}^* : x \text{ starts and ends with the same character}\}$ can be accepted by a DFA $M = (Q, \Sigma, \delta, q_0, F)$ for which $\delta^*(q_0, w) = q_0$ for some $w \neq \varepsilon$. Assume an alphabet of $\Sigma = \{a, b\}$.

Problem 4 A *homomorphism* is a function $h : \Sigma \to \Gamma^*$ for alphabets Σ, Γ. Given a homomorphism h, extend it to strings and then languages by asserting that $h(\varepsilon) = \varepsilon$, $h(a_1 \cdots a_n) = h(a_1) \cdots h(a_n)$ (for $a_1, \ldots, a_n \in \Sigma$), and $h(L) = \{h(x) : x \in L\}$.

(a) Prove: for any homomorphism h, if L is DFA-acceptable, then so is $h(L)$.

(b) Disprove: for any homomorphism h, if $h(L)$ is DFA-acceptable, then so is L.

Problem 5 Prove that five states are necessary and sufficient to recognize the language $L = \{x \in \{a,b\}^+ : x \text{ starts and ends with the same character}\}$.

Problem 6. As mentioned in class, the *names* of states—what we choose as points for the set Q—is unimportant for defining what **are** DFA-acceptable languages. Re-formalize the syntax of a DFA so that there is no named state set (use consecutive numbers instead). With the new convention, a DFA will be some sort of 4-tuple. Give the formal definition for each of its components. Then describe something we have done in class that will look **more** complex using your new formulation.