Problem Set 2 – Due Thursday, October 11, 2012

Problem 1 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 $\delta^*(q, ax) = \delta^*(\delta(q, a), x)$. Carefully justifying each step, prove that $\delta^*(q, xy) = \delta^*(\delta^*(q, x), y)$.

Problem 2 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 and alphabet of $\Sigma = \{a, b\}$.

Problem 3 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 4 Recall the DIOPHANTINE EQUATION problem: given a multivariate polynomial P with integer coefficients, decide whether or not P has an integer root.

(a) In the first lecture I claimed without proof that there is no algorithm to answer this question. But suppose I provide you with an *oracle* (a "magic box") that did answer the question. In a single computational step, it says *yes* or *no* according to whether or not P has an integer root. Given such an oracle, describe an algorithm that *finds* an integer root of any multivariate polynomial that has one (and reports *No Root* otherwise).

(b) Let $s(n)$ be the maximum number of computational steps that your algorithm takes to run (on some fixed, oracle-containing computer) when the polynomial P is described by a string of length n. Explain why there is no algorithm to compute $S(n)$ for *any* function S such that $S(n) \geq s(n)$ for all n. (In brief, some functions just grow *too fast* to be computable.)

Problem 5 Let $\mathit{Extend}(L) = \{x \in L : \text{ there exists a } y \in \Sigma^+ \text{ for which } xy \in L\}$.

(a) What is $\mathit{Extend}(\{\mathtt{0}, \mathtt{1}\}^*)$? What is $\mathit{Extend}(\{\varepsilon, 0, 1, 00, 01, 111, 1110, 1111\})$?

(b) Prove that if L is DFA-acceptable then $\mathit{Extend}(L)$ is too.