CS154: Introduction to Automata and Complexity Theory **Fall 1997**

Assignment #1

Due: Wednesday, October 8th, 1997.

Problem 1 (3 points) Given two regular languages L_1, L_2 we construct the *product automaton* that recognizes their intersection $L_1 \cap L_2$. If $M_1 = (Q_1, \Sigma, \delta_1, q_0^{(1)}, F_1)$ recognizes L_1 and $M_2 = (Q_2, \Sigma, \delta_2, q_0^{(2)}, F_2)$ recognizes L_2 then the product automaton $M = (Q, \Sigma, \delta, q_0, F)$ is defined as:

$$\begin{aligned} Q &= Q_1 \times Q_2 \\ q_0 &= (q_0^{(1)}, q_0^{(2)}) \\ F &= \{(q_1, q_2) \mid q_1 \in F_1 \text{ and } q_2 \in F_2\} \\ \delta\Big((q_1, q_2), a\Big) &= \Big(\delta_1(q_1, a), \delta_2(q_2, a)\Big) \end{aligned}$$

Formally prove the following claim: the automaton M recognizes $L_1 \cap L_2$. To get full credit you must justify every step in your proof.
Hint: Structure your proof as follows:

1. To prove $L(M) = L_1 \cap L_2$ first prove that $L(M) \subseteq L_1 \cap L_2$ then prove $L(M) \supseteq L_1 \cap L_2$.
2. To prove $L(M) \subseteq L_1 \cap L_2$ show that every word x accepted by M is in $L_1 \cap L_2$. Recall that a word is accepted by M if there exists a sequence of states
$$(q_1^{(1)}, q_2^{(1)}) \ , \ (q_1^{(2)}, q_2^{(2)}) \ , \ \ldots \ , \ (q_1^{(n)}, q_2^{(n)}) \ \in \ Q$$
satisfying the three conditions of an *accepting computation*. Show that this sequence of states implies that both M_1 and M_2 must accept x. That is, construct two sequences of states that are accepting computations for M_1 and M_2.
3. To prove $L(M) \supseteq L_1 \cap L_2$ show that every $x \in L_1 \cap L_2$ is accepted by M. Use the fact that $x \in L_1 \cap L_2$ to construct an accepting computation for M. Prove that the sequence of states you construct indeed satisfies the three conditions for being an *accepting computation*.

Problem 2 (2 points)

1. Design a deterministic automaton that accepts all binary words that *do not* contain two consecutive zeros. A diagram is sufficient. The fewer states you use the better.
Hint:there exists an automaton with only three states.
2. Build the regular expression for the language from part (1).

Problem 3 (2 points) Given a deterministic automaton $M = (Q, \Sigma, \delta, q_0, F)$ accepting the language L describe the language accepted by the automaton $M' = (Q, \Sigma, \delta, q_0, Q \setminus F)$ where $Q \setminus F$ stands for *set minus*, i.e. $Q \setminus F = \{q \mid q \in Q \text{ and } q \notin F\}$.

Problem 4 (3 points) For a word w define w^r to be the reversed word. That is, if $w = w_1 w_2 \ldots w_n$ then $w^r = w_n w_{n-1} \ldots w_2 w_1$. Let L be a regular language.
Prove that $L^r = \{x^r \mid x \in L\}$ is also a regular language.
Hint: Given an automaton $M = (Q, \Sigma, \delta, q_0, F)$ accepting L formally construct an automaton $M' = (Q', \Sigma, \delta', q_0', F')$ accepting L^r.