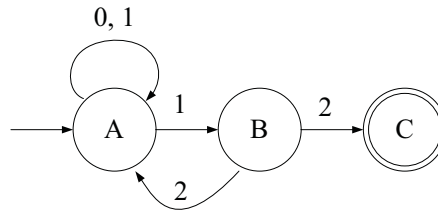


Problem Set 3 Solutions

Problem 1. Using the procedure shown in class, convert the following NFA into a DFA for the same language.



This is a pretty mechanical problem; see me or the TA if you had trouble with it.

Problem 2. For any language L let

$$\text{noprefix}(L) = \{w \in L \mid \text{no proper prefix of } w \text{ is a member of } L\}$$

Prove or disprove: if L is DFA-acceptable then so is $\text{noprefix}(L)$.

This is true; we prove it constructively, showing that if L is DFA acceptable then $\text{noprefix}(L)$ is NFA-acceptable. This is sufficient because NFA-acceptable languages are also DFA-acceptable, by the subset construction. Let $M = (Q, \Sigma, \delta, q_0, F)$ be a DFA such that $L(M) = L$. We construct an NFA $M' = (Q, \Sigma, \delta', q_0, F)$ such that $L(M') = \text{noprefix}(L)$. The construction is simple: just sever each transition going out of each final state. Formally, $\delta'(q, a) = \{\delta(q, a)\}$ if $q \in Q - F$, while $\delta'(q, a) = \emptyset$ for $q \in F$, and $\delta'(q, \varepsilon) = \emptyset$ for each $q \in Q$.

Now why does the above construction work? Suppose $x \in L(M')$. Let x' be any proper prefix of x . By construction, when M' runs on x' , M' must terminate in a non-final state of M (because had we ended in a final state of M , we would not have been able to progress out of that state in M' , contradicting the fact that $x \in L(M')$). If M processes x' , we end up in that same non-final state. Thus $x' \notin L(M)$. We conclude that $L(M') \subseteq \text{noprefix}(L)$.

On the other hand, suppose $x \in \text{noprefix}(L)$. Then as x is run on the DFA M , we never pass through a final state of M until the very last character of x is consumed. So the exact same sequence of transitions exist in M' and provide an accepting path of M' . Therefore $\text{noprefix}(L) \subseteq L(M')$.

Problem 3. For $n \geq 0$, let $L_n = \{1^i : 0 \leq i < n\}$ (where $1^0 = \varepsilon$). Prove that there is a DFA M_n having n final states that accepts L_n . Then prove that L_n cannot be accepted by any DFA having fewer accept states.

This problem was postponed for one week; solution to be supplied next week.

Problem 4. Consider applying the product construction to NFAs $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ and $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$ in order to show that the NFA-acceptable languages are closed under intersection.

Part A. Formally specify the product machine $M = (Q, \Sigma, \delta, q_0, F)$.

First, assume that every state has an ε -arrow to itself: $q \in \delta_1(q, \varepsilon)$ for each $q \in Q_1$ and $q \in \delta_2(q, \varepsilon)$ for each $q \in Q_2$. (Figure out what goes wrong in what's below if I don't say this.) Then define:

$$Q = Q_1 \times Q_2$$

$$\delta((p, q), \sigma) = \delta_1(p, \sigma) \times \delta_2(q, \sigma) \text{ for each } p \in Q_1, q \in Q_2, \text{ and } \sigma \in \Sigma \cup \{\varepsilon\}$$

$$q_0 = (q_1, q_2)$$

$$F = F_1 \times F_2$$

Part B. Does the construction work—that is, is $L(M) = L(M_1) \cap L(M_2)$? Informally argue your conclusion.

Yes, the construction does work. Suppose that $x \in L(M_1) \cap L(M_2)$. Then, for some $s_1 \in F_1$, there is some $q_1 \rightarrow s_1$ path in M_1 labeled by x , and for some $s_2 \in F_2$, there is some $q_2 \rightarrow s_2$ path in M_2 labeled by x . One constructs an x -labeled $(q_1, q_2) \rightarrow (s_1, s_2)$ path in M in the natural way, by combining the two paths. Similarly, suppose $x \in L(M)$. Then for some $(s_1, s_2) \in F_1 \times F_2$ there is an x -labeled $(q_1, q_2) \rightarrow (s_1, s_2)$ path in M . From this path one can read off an x -labeled $q_1 \rightarrow s_1$ path in M_1 and an x -labeled $q_2 \rightarrow s_2$ path in M_2 . So $x \in L(M_1) \cap L(M_2)$.