**Solutions for Problem Set 2**  
CS 373: Theory of Computation  

Assigned: September 7, 2010  
Due on: September 14, 2010 at 10am

Homework Problems

**Problem 1.** [Category: Design] Design an NFA for the language $D$ given in Problem 1.48. You need not formally prove the correctness of your construction, but your construction should be clear and understandable.

**Solution:** Language $D$ is the collection of all binary strings that have an equal number of 01 and 10 occurrences. Observe that we have one occurrence of 01 when there is switch from a sequence of 0s to a sequence of 1s. Similarly there is an occurrence of 10 when there is a switch from a sequence of 1s to a sequence of 0s. Thus, since the runs of 0s and 1s alternate in a binary string, a string has equal number of 0s and 1s iff the strings begins and ends in the same symbol. This yields the following NFA. Formally, the

\[
NFA \; N = (\{q_0, q_1, q_2, q_3, q_4\}, \{0, 1\}, \delta, q_0, \{q_0, q_2, q_4\})
\]

where $\delta$ is given by

\[
\begin{align*}
\delta(q_0, 0) &= \{q_1, q_2\} \\
\delta(q_1, 0) &= \{q_1, q_4\} \\
\delta(q_2, 0) &= \{q_3\} \\
\delta(q_3, 0) &= \{q_3, q_4\}
\end{align*}
\]

For all other values of $q$ and $a \in \{0, 1\} \cup \{\epsilon\}$, $\delta(q, a) = \emptyset$.

**Problem 2.** [Category: Design] Problem 1.31. *Hint:* Show that if $M$ is a DFA recognizing $A$ then there is an NFA recognizing $A^R$.

**Solution:** Given a DFA $M = (Q, \Sigma, \delta, q_0, F)$ recognizing $A$, the NFA $N$ recognizing $A^R$ will be obtained by “reversing” the transitions and switching the initial and final states. Formally, $N = (Q', \Sigma, \delta', q_0', F')$ where
• $Q' = Q \cup \{q'_0\}$, where $q'_0 \notin Q$

• $F' = \{q_0\}$

• $\delta'$ is given by

$$
\delta'(q, a) = \begin{cases} 
F & \text{if } q = q'_0 \text{ and } a = \epsilon \\
\{q' \in Q \mid \delta(q', a) = q\} & \text{if } q \in Q \\
\emptyset & \text{otherwise}
\end{cases}
$$

Correctness of the construction will be established by showing that for $q \in Q$ (i.e., states of DFA $M$), and any $w \in \Sigma^*$

$$\hat{\delta}(q_0, w) = q \text{ if and only if } q_0 \in \hat{\Delta}(q, w^R)$$

Assuming that the above claim holds, establish the correctness of the construction follows from the below reasoning.

- $w$ is accepted by $M$ if and only if $q_0 \in \hat{\Delta}(q_0, w)$

The claim is proved by induction on the length of $w$.

- **Base Case:** Observe that $\hat{\delta}(q_0, \epsilon) = q_0$ and $q_0 \in \hat{\Delta}(q_0, \epsilon)$. Thus, the claim holds for the base case.

- **Induction Step:** Assume that the claim holds for all strings $u$ of length $n$. Consider a string $w = ua$ of length $n + 1$, where $a \in \Sigma$ and $u$ is of length $n$. Taking $\hat{\delta}(q_0, w) = q$ and $\hat{\delta}(q_0, u) = q'$, we have

$$
q = \hat{\delta}(q_0, ua) \quad \text{if } q = \delta(q', a) \\
\text{if } q' \in \delta'(q, a) \text{ and } q_0 \in \hat{\Delta}(q', u^R) \\
\text{if } q_0 \in \hat{\Delta}(q, au^R) \\
\text{if } q_0 \in \hat{\Delta}(q, u^R)
$$

$$
(\hat{\delta}(q, u_a) = \delta(\hat{\delta}(q, u), a)) \\
(\text{defn. of } \delta' \text{ and ind. hyp. on } u) \\
(\text{prop. } \hat{\Delta}(q, uv) = \cup_{q' \in \hat{\Delta}(q, u)} \hat{\Delta}(q', v)) \\
(w^R = au^R)
$$

**Problem 3. [Category: Comprehension+Proof] Read problem 1.38.**

(a) Give formal definitions of all-NFA, acceptance of a string $w$, and language recognized.

(b) Prove that $L$ is regular if and only if there is an all-NFA $A$ such that $L(A) = L$.

**Solution:**

(a) Formally, an all-NFA is $M = (Q, \Sigma, \delta, q_0, F)$, where $Q$ is a finite set of states, $\Sigma$ is a finite input alphabet, $\delta : Q \times (\Sigma \cup \{\epsilon\}) \to 2^Q$ the transition function, $q_0 \in Q$ the initial state, and $F \subseteq Q$ the set of final/accept states. As for NFAs, define $q \xrightarrow{w_M} q_2$ if and only if there exist $r_0, r_1, \ldots, r_k$ and $x_1, x_2, \ldots, x_k$ such that $r_i \in Q$, $x_i \in \Sigma \cup \{\epsilon\}$, and
1. \( r_0 = q_1 \) and \( r_k = q_2 \).
2. \( w = x_1 x_2 \cdots x_k \), and
3. for all \( i \in \{1, \ldots, k\} \), \( r_i \in \delta(r_{i-1}, x_i) \)

Let us define \( \hat{\Delta}(q, w) = \{ q' \in Q \mid q \xrightarrow{w}^M q' \} \). Then \( w \) is accepted by \( M \) iff \( \hat{\Delta}(q_0, w) \neq \emptyset \) and \( \hat{\Delta}(q_0, w) \subseteq F \); note that acceptance requires that at least one thread be active and that all the active threads are in accepting states. Finally, \( L(M) = \{ w \in \Sigma^* \mid \hat{\Delta}(q_0, w) \subseteq F \} \).

(b) (⇒) Suppose \( L \) is regular. In other words, there is a DFA \( D = (Q, \Sigma, \delta, q_0, F) \) such that \( L(D) = L \). Consider \( N = (Q, \Sigma, \delta', q_0, F) \) where \( \delta'(q, a) = \{ \delta(q, a) \} \), when \( a \in \Sigma \), and \( \delta'(q, \epsilon) = \emptyset \). We show that \( L(N) = L(D) \). To prove this we establish the following stronger claim by induction on the length of \( w \): for every \( q \in Q \) and \( w \in \Sigma^* \), \( \hat{\Delta}(q, w) = \{ \delta(q, w) \} \). Observe that if the claim is established, \( L(N) = L(D) \), because \( w \in L(D) \) iff \( \hat{\delta}(q_0, w) \in F \) iff \( \hat{\Delta}(q_0, w) \subseteq F \) (by claim) iff \( w \in L(N) \).

To establish the claim in the base case, observe that because of the definition of \( \delta' \), we have \( \hat{\Delta}(q_0, \epsilon) = \{ q_0 \} = \{ \delta(q_0, \epsilon) \} \). For the induction step, assume that \( \hat{\Delta}(q, w) = \{ \delta(q, w) \} \) for strings \( w \) of length \( n \). Consider \( w = ua \) of length \( n + 1 \), where \( u \in \Sigma^* \) of length \( n \) and \( a \in \Sigma \). Now,

\[
\hat{\Delta}(q_0, ua) = \bigcup_{q \in \Delta(q_0, u)} \hat{\Delta}(q, a) \quad \text{(defn. of } \hat{\Delta})
\]

\[
= \bigcup_{q \in \Delta(q_0, u)} \delta'(q, a) \quad \text{(as } N \text{ does not have any } \epsilon\text{-transitions)}
\]

\[
= \delta'(\delta(q_0, u), a) \quad \text{(ind. hyp.)}
\]

\[
= \{ \delta(q_0, ua) \} \quad \text{(defn. of } \delta' \text{)}
\]

\[
= \{ \delta(q, w) \} \quad \text{(defn. of } \delta \text{)}
\]

(⇐) Let \( N = (Q, \Sigma, \delta, q_0, F) \) be a all-NFA. The construction of a DFA \( D \) that recognizes the same language as \( N \) is the same as that of a DFA equivalent to an NFA, except for the final states. The correctness proof is also very similar. The details are as follows. The DFA \( D = (2^Q, \Sigma, \delta', q_0', F') \) where

- \( q_0' = \hat{\Delta}(q_0, \epsilon) \)
- \( F' = \{ A \subseteq Q \mid A \neq \emptyset \text{ and } A \subseteq F \} \); note requiring that the emptyset not be a final state is a very important.
- \( \delta'(A, a) = \cup_{q \in A} \hat{\Delta}(q, a) \)

The correctness of the construction will follow by establishing that for all strings \( w \in \Sigma^* \), \( \hat{\Delta}(q_0, w) = \hat{\delta}(q_0', w) \). This is because, \( w \in L(D) \) iff \( \hat{\delta}(q_0', w) \in F' \) iff \( \hat{\delta}(q_0', w) \neq \emptyset \) and \( \hat{\delta}(q_0', w) \subseteq F \) iff \( \hat{\Delta}(q_0, w) \neq \emptyset \) and \( \hat{\Delta}(q_0, w) \subseteq F \) (by claim) iff \( w \in L(N) \).

The claim that for all \( w \in \Sigma^* \), \( \hat{\Delta}(q_0, w) = \hat{\delta}(q_0', w) \) will be established by induction on the length of \( w \). For the base case, observe that \( \hat{\delta}(q_0', \epsilon) = q_0' = \hat{\Delta}(q_0, \epsilon) \). In the induction hypothesis, we assume that the claim holds for all \( w \) of length \( n \). For the induction step, consider \( w = ua \), where \( u \) is of length \( n \) and \( a \in \Sigma^* \). Now,

\[
\hat{\delta}(q_0', ua) = \delta'(\hat{\delta}(q_0', u), a) \quad \text{(defn. of } \hat{\delta})
\]

\[
= \delta'(\hat{\Delta}(q_0, u), a) \quad \text{(ind. hyp. on } u \text{)}
\]

\[
= \bigcup_{q \in \hat{\Delta}(q_0, u)} \hat{\Delta}(q, a) \quad \text{(defn. of } \delta' \text{)}
\]

\[
= \hat{\Delta}(q_0, ua) \quad \text{(prop } \hat{\Delta}(q, uw) = \bigcup_{q' \in \hat{\Delta}(q, u)} \hat{\Delta}(q', v) \text{)}
\]

\[\blacksquare\]