Homework Assignment 5

Due: Thursday, March 17, 2005, by 12 PM (IN CLASS)

Please, read the following instructions:

• Fill out this form with your name, student ID, email, and signature and return it as the cover page of your homework.

• Turn in your homework at the beginning of your class on the due date described at the top of this page.

• Late assignments will be penalized 25% and will not be accepted after 1:30PM of the day following the due date.

• Late assignments must be turned in to Janean Williams in room 308, 3rd floor, Levine Building.

• All writings must be neat, well-organized, and include sufficient explanations in the delineation of the solutions.

• Full credit will be given only in the case where the correct answer has been properly justified with complete explanations.

• Good luck!

<table>
<thead>
<tr>
<th>Question</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max</td>
<td>15</td>
<td>25</td>
<td>20</td>
<td>25</td>
<td>15</td>
</tr>
<tr>
<td>Score</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TOTAL:_________________________________________
Problem 1 [15 points]

Let $\Sigma = \{a, b, c\}$. Then,

(a) Give a regular expression that describes the set of all strings $w$ in $\Sigma^*$ such that $|w|$ is odd and $w$ ends with a $b$.

(b) Give a regular expression that describes the set of all strings $w$ in $\Sigma^*$ such that $w$ do not contain two consecutive $c$’s.

Solution:

(a) $\left((a + b + c)(a + b + c)\right)^*b$

(b) $(a + b + c(a + b))^*(\epsilon + c)$
Problem 2 [25 points]

We saw in class that regular languages are closed under complementation, union, concatenation, Kleene closure, and intersection. Use one of these facts and the given fact that the language

\[ L_1 = \{ w \in \{0, 1\}^* \mid w = 0^n1^n \text{ for some non-negative integer } n \} \]

is not regular to prove that the language

\[ L_2 = \{ w \in \{0, 1\}^* \mid w = 0^j1^k \text{ for some non-negative integers } j, k \text{ such that } j \neq k \} \]

is also not regular.

Solution:
The key to solve this question is to realize the relationship between \( L_1 \) and \( L_2 \). First, note that \( L_1 \cap L_2 = \emptyset \) and \( L_1 \cup L_2 = L(0^*1^*) = \{0\}^*\{1\}^* \). This means that

\[ L_1 = \{a\}^*\{b\}^* - L_2. \]

The expression below is not very helpful, as it only says to us that \( L_1 \) is the complement of \( L_2 \) with respect to the language \( \{a\}^*\{b\}^* \), and hence we cannot use any of the properties of regular languages we have learned. However, thinking a bit longer, you can realize that

\[ L_1 = \{a\}^*\{b\}^* - L_2 = \{a\}^*\{b\}^* \cap (\{a, b\}^* - L_2). \]

Now, we are in business, as we know something about intersection and complementation of regular languages!

How can we use our knowledge? The trick is to prove our claim by contradiction. Assume that \( L_2 \) is regular. So, since the complementation of a regular language is a regular language, we have that \( \{a, b\}^* - L_2 \) is a regular language. Since \( \{a\}^*\{b\}^* \) is a regular language and so is \( \{a, b\}^* - L_2 \), we must have that \( \{a\}^*\{b\}^* \cap (\{a, b\}^* - L_2) \) is also a regular language, as the intersection of two regular languages is a regular language. But, since \( \{a\}^*\{b\}^* \cap (\{a, b\}^* - L_2) = L_1 \) and \( L_1 \) is not regular (from the statement of the problem), our assumption that \( L_2 \) is regular leads us to a contradiction. So, \( L_2 \) cannot be regular.
Problem 3 [20 points]

Let $\Sigma$ be an alphabet, and let $L$ be any language over $\Sigma$. Then, we define the unary operator $E$ as

$$E(L) = \{ w \in \Sigma^* \mid w = xy, \text{ where } x \in \Sigma^+ \text{ and } y \in L \}.$$ 

(a) Given a DFA accepting $L$, describe how to modify this DFA to create another DFA (or an NFA, or an $\epsilon$-NFA) accepting $E(L)$.

(b) Assuming that your construction for (a) is correct, argue that the set of all regular languages closed under the operator $E$.

Solution:

(a) Let $D = (Q, \Sigma, \delta, q_0, F)$ be a DFA such that $L(D) = L$. We modify the DFA $D$ to construct an $\epsilon$-NFA $N$ in two steps. First, we build another DFA $D_1 = (Q_1, \Sigma, \delta_1, q_{0_1}, F_1)$, such that $Q_1 = \{ r_1, r_2 \}$, $q_{0_1} = r_1$, $F_1 = \{ r_2 \}$, and the transition function $\delta_1 : Q_1 \times \Sigma \rightarrow Q_1$ is defined as $\delta_1(r_1, a) = r_2$ and $\delta_1(r_2, a) = r_2$, for all $a \in \Sigma$. That is, $L(D_1) = \Sigma^+$. Second, we build an $\epsilon$-NFA $N = (Q_N, \Sigma, \delta_N, q_{0_N}, F_N)$ concatenating $D_1$ with $D$ using the construction we saw in class: $Q_N = Q_1 \cup Q$, $q_{0_N} = r_1$, $F_N = F$, and the transition function $\delta_N : Q_N \times (\Sigma \cup \{ \epsilon \}) \rightarrow P(Q_N)$ is defined as $\delta_N(s, a) = \{ \delta_1(s, a) \}$ for all $s \in Q_1$ and $a \in \Sigma$, $\delta_N(s, \epsilon) = \{ \delta(s, \epsilon) \}$ for all $s \in Q$ and $a \in \Sigma$, and $\delta_N(r_2, \epsilon) = \{ q_{0_1} \}$ and $\delta_N(s, \epsilon) = \emptyset$ for all $s \in (Q_N - \{ r_2 \})$. So, by construction, the language $L(N)$ recognized by our $\epsilon$-NFA $N$ is precisely $L(D_1)L(D) = \Sigma^+L$. Now, note that $E(L) = \Sigma^+L$ by definition!

(b) If our construction for (a) is correct, then for any regular language $L$, the language $E(L)$ is also regular, as we can build an $\epsilon$-NFA that accepts $E(L)$. So, the set of all regular languages over $\Sigma$ is indeed closed under the operator $E$. 
Problem 4 [25 points]

Let $L$ be a language over an alphabet $\Sigma$, and let $a$ be a symbol of $\Sigma$. We define the quotient $L/a$ of $L$ and $a$ to be the language of all strings $w$ in $\Sigma^*$ such that $wa$ is in $L$. For example, if $\Sigma = \{0, 1\}$ and $L = \{0, 001, 100\}$ then $L/0 = \{\epsilon, 10\}$.

(a) Given a DFA accepting $L$, describe how to modify this DFA to create another DFA (or an NFA, or an $\epsilon$-NFA) accepting $L/a$.

(b) Prove that the finite state machine you provided as an answer for (a) is correct.

Solution:

(a) Let $D = (Q, \Sigma, \delta, q_0, F)$ be a DFA such that $L(D) = L$. We build another DFA $D_1 = (Q, \Sigma, \delta, q_0, F_1)$ such that the set $F_1$ of final states of $D_1$ is as follows: For every $q \in Q$, the state $q$ is in the set $F_1$ of final states of $D_1$ if and only if there exists a transition from $q$ to a final state of $F$ on the symbol $a$, i.e., if and only if the state $\delta(q, a)$ is a final state of $D$. In other words, the DFA’s $D$ and $D_1$ are identical, except (perhaps) for the set of final states $F$ and $F_1$.

(b) To prove that our answer for (a) is correct means to prove that $L(D_1) = L(D)/a$. Since both $L(D_1)$ and $L(D)/a$ are sets, we must prove that $L(D_1) \subseteq L(D)/a$ and $L(D)/a \subseteq L(D_1)$. Let us show that $L(D_1) \subseteq L(D)/a$. Let $w$ be any string from $L(D_1)$. Since $w \in L(D_1)$, we know that $\delta(q_0, w) = q$, for some $q \in F_1$. From the construction of $D_1$, we must have that $\delta(q, a) \in F$. So, the string $wa \in L(D)$, and by definition of $L(D)/a$, the string $w$ belongs to $L(D)/a$. So, $L(D_1) \subseteq L(D)/a$. Now, let us show that $L(D)/a \subseteq L(D_1)$. Let $w$ be any string from $L(D)/a$. From the definition of $L(D)/a$, if $w \in L(D)/a$ then $wa \in L(D)$. So, $\delta(q_0, wa) \in F$. Since $\delta(q_0, wa) = \delta(\delta(q_0, w), a) = \delta(q, a)$, for some $q \in Q$ such that $q = \delta(q_0, w)$. From the construction of $D_1$, we know that $q \in F_1$ if and only if $\delta(q, a) \in F$. Since $\delta(q, a) \in F$ is true, we must have $q \in F_1$, and therefore $\delta(q_0, w) \in F_1$, which means that $w \in L(D_1)$. So, $L(D)/a \subseteq L(D_1)$, and we can conclude that $L(D_1) = L(D)/a$.

Note that our proof was made easier by the fact that $D$ and $D_1$ are basically the same DFA, except (perhaps) for the sets $F$ and $F_1$. This means that $D$ and $D_1$ have the same transition function, and this rules out the need for an inductive proof as the one in a similar problem given in Recitation 6. The lesson one should learn from this problem is that one should always try to verify the best method of proof to solve a given problem. Sometimes, a given method makes the solution harder or even impossible, while another method makes it easier or less difficult.
Problem 5 [15 points]

Let $\mathcal{N}_{\Sigma}$ be the set of all languages over an alphabet $\Sigma$ that are not regular. Is $\mathcal{N}_{\Sigma}$ closed under union? If your answer is yes, prove it. Otherwise, give a counterexample.

Solution:

No, the set $\mathcal{N}_{\Sigma}$ is not closed under union. For a counterexample, consider the languages $L_1$ and $L_2$ in Problem 2. Both $L_1$ and $L_2$ are not regular, which means that $L_1$ and $L_2$ are in $\mathcal{N}_{\Sigma}$. However, $L_1 \cup L_2 = \{a\}^*\{b\}^*$, which is clearly a regular language. So, $L_1 \cup L_2 = \{a\}^*\{b\}^*$ does not belong to $\mathcal{N}_{\Sigma}$, and hence $\mathcal{N}_{\Sigma}$ is not closed under union.