Problem Set 3

Spring 10

Due: Thursday Feb 18 in class before the lecture.

Please follow the homework format guidelines posted on the class web page:

`http://www.cs.uiuc.edu/class/sp10/cs373/`

1. [**Category**: Comprehension, **Points**: 20]

   Give a regular expression for the following languages:

   (a) $\Sigma = \{a, b\}$: The set of all strings where the second letter from the start and second letter from the end both are a's (e.g. $babaab$).

   (b) $\Sigma = \{a, b\}$: The set of all strings that have both aa and bb as a (contiguous) substring.

   (c) $\Sigma = \{a, b, c\}$: The set of all strings, such that between every a and c there's at least one b.

   Describe the language of each of the following regular expressions in your own words. Please be specific and try to minimize the amount of mathematical notation you use.

   (a) $\Sigma = \{a, b\}$. $(ab + ba)*$

   (b) $\Sigma = \{a, b\}$. $((a^*)b(a^*)b(a^*))^*b$

   (c) $\Sigma = \{a, b, c\}$. $((\epsilon + a + aa + aaa)(b + c))^*(\epsilon + a + aa + aaa)$

2. Intersect 'em [**Category**: Construction, **Points**: 20]

   You are given two NFAs $A_1 = (P, \Sigma, \delta_1, p_0, F_1)$ and $A_2 = (Q, \Sigma, \delta_2, q_0, F_2)$.

   Construct an NFA that will accept the language $L(A_1) \cap L(A_2)$ with no more than $|P| * |Q|$ states. Also, prove that it indeed accepts the language of the intersection as stated above.

   This question requires a formal construction using tuple-notation and a proof.

3. Reverse determinism [**Category**: Construction, **Points**: 20] (EXTRA CREDIT/HONORS)

   This problem is due on Feb 25 in class.

   Recall the formal definition of an NFA (Sipser p. 53). Let us generalize the definition by substituting the unique start state q_0 by a *set* of initial states S, so that the computation of an NFA is allowed to start from any state in S. A *Reverse Deterministic Automaton*(RDA) is an generalized NFA $A = (Q, \Sigma, \delta, S, F)$ where

   (a) for each state $q \in Q$, $\delta(q, \epsilon) = \varnothing$; (i.e. there are no epsilon transitions).

   (b) for each state $q \in Q$ and each letter $x \in \Sigma$, there is *at most one state* $p \in Q$ such that $q \in \delta(p, x)$;

(c) $|F| = 1$.

Graphically, an RDA does not allow two distinct states to transition into the same state via two transitions reading the same input letter. Moreover, an RDA has multiple start states, a unique accept state, and no ϵ-transitions.

Given an RDA $A = (Q, \Sigma, \delta, S, q_f)$, construct an RDA $\overline{A}$ with no more than $|Q| + 1$ states that will accept the complement language $\overline{L(A)}$. Prove that $\overline{A}$ is indeed an RDA and accepts the complement of $L(A)$.

(This question requires a formal construction and a proof.)