CSE 322: Formal Models in Computer Science — **Assignment #3**

April 18, 2008 — Due: **Friday**, April 25, 2008

Reading Assignment: Sipser 1.4

Problems:

1. For the language denoted by each of the following regular expressions, give two strings that are members and two strings that are non-members - a total of four strings for each part. Assume the alphabet $\Sigma = \{a, b\}$ in all parts:
   (a) $a(ab)^*b$
   (b) $a^* \cup b^*$
   (c) $\Sigma^* a \Sigma^* b \Sigma^* a \Sigma^*$
   (d) $(aba \cup bab) \cup \emptyset$
   (e) $(\varepsilon \cup a)b$
   (f) $(a \cup ba \cup bb)\Sigma^*$

2. Give regular expressions for the following languages:
   (a) L_3 , which is the language of all valid comments in the C language. Assume for this problem that a valid comment in C starts with /# and ends with #/ with no intervening #/. Assume for simplicity that the alphabet for L_3 is $\Sigma = \{/, \#, a, b\}$. For example, $/\#ab\#/$ and $/\#a/\#b\#/$ are in L_3 while $/\#ab$ and $/\#a\#/b\#/$ are not.
   (b) $L_4 = \{w | w \in \{0,1\}^*$ and w starts with a 1 and has odd length or w starts with a 010 and has length that is a multiple of 3$\}$.

3. Using the construction given by the proof of Lemma 1.55 (1st edition Lemma 1.29) (as shown in Examples 1.57 and 1.58 (1st edition 1.30 and 1.31) to draw state diagrams for NFAs that accept the languages given by the following regular expressions. Include all states that would be created by this construction (in other words do not simplify any of the steps in the construction.)
   (a) $((ab)^*b)^* \cup bb^*$.
   (b) $(a^*b(ab^*)^*b^*)^*$

4. Consider the DFA $M = (\{q_0, q_1, q_2, q_3\}, \{0,1\}, \delta, q_0, \{q_1\})$ with the following transition function for $i = 0, 1, 2, 3$; $\delta(q_i, 0) = q_{2i \bmod 4}$ and $\delta(q_i, 1) = q_{2i+1 \bmod 4}$.
   (a) Let L_5 be the language accepted by M. Give a simple description of L_5.
   (b) Using the NFA to regular expression covered in class, obtain a regular expression that describes L_5. Please show your steps. You may make obvious reductions in the regular expressions on the GNFA as you proceed.

5. **Extra Credit** Do it for the glory glory glory, not for the points (no tripling of the word points, because the bonus points are minimal.) Prove that every regular language is accepted by a planar NFA. An NFA is planar if it can be embedded in the plane (i.e. one can draw it) such that there are no crossings.