CSC 236 November 17, 2004
St George Campus University of Toronto

Assignment #5
Due: December 2, 2004, by 6:00 PM
(in the CSC 236 drop box in BA 2220)

Question 1. (10 marks) State whether each of the following statements is true ***for all*** regular expressions R and S. Justify your answers.

a. (3 marks) $(R+S)^* \equiv (R^*S^*)^*$

b. (3 marks) $(R+S)^* \equiv R^* + S^*$

c. (4 marks) $(RS+R)^*R \equiv R(SR+R)^*$

Question 2. (10 marks) For each of the languages below, give a DFSA that accepts the language and a regular expression that denotes it. For each DFSA and regular expression, give an ***informal but informative*** argument to justify its correctness.

$$L = \{x \in \{0,1\}^* : \text{neither 00 nor 11 is a substring of } x\}$$
$$L' = \{x \in \{0,1\}^* : \text{both 00 and 11 are substrings of } x\}$$

Question 3. (10 marks)

a. For any language L, define $\mathbf{Odd}(L) = \{x : x \in L \text{ and } |x| \text{ is odd}\}$. Prove that if L is accepted by a FSA, then $\mathbf{Odd}(L)$ is also accepted by a FSA.

b. For any languages L, L', we define the "shuffle" operation $\bowtie$ as follows:

$L \bowtie L' = \{x \in \Sigma^* :$ either $x = \epsilon$,
or there is a positive integer k and strings $y_1, y_2, \ldots, y_k \in L$ and $y'_1, y'_2, \ldots, y'_k \in L'$
so that $x = y_1 y'_1 y_2 y'_2 \cdots y_k y'_k\}$

Prove that if each of L, L' is accepted by a FSA then $L \bowtie L'$ is also accepted by a FSA.

Question 4. (15 marks) Define the following language over the alphabet $\{1,2,3\}$:

$$L_3 = \{x \in \{1,2,3\}^* : \text{ at least one symbol appears an odd number of times in } x\}.$$

a. (5 marks) Give a nondeterministic FSA that accepts L_3. To make part (b) easier, you should construct a NFSA with as few states as possible. (It is relatively easy to do it with seven states, and I know it is possible to do it with six. I don't know if it can be done with fewer.) Informally explain why your NFSA is correct.

b. (5 marks) Apply the subset construction to the NFSA of part (a) and show the resulting DFSA.

c. (5 marks) Prove that no DFSA that accepts L_3 has fewer than eight states. (**Hint:** Prove that, for each $a \in \{1,2,3\}$, if x and x' are strings in $\{1,2,3\}^*$ that differ (at least) in the parity of the number of occurrences of a, then $\delta^*(s,x) \neq \delta^*(s,x')$, where s is the start state and δ is the transition function of any DFSA that accepts L_3.)

For further thought—this question is not part of the assignment. Generalise the above results as follows. For any $n \geq 2$, let L_n be the following language over the alphabet $\{1, 2, \ldots, n\}$:

$$L_n = \{x \in \{1, 2, \ldots, n\}^* : \quad \text{at least one symbol appears an odd number of times in } x\}.$$

Prove that

(i) there is a NFSA with only $2n+1$ states that accepts L_n (actually it's possible to prove that $2n$ states suffice, but the slightly weaker result of $2n+1$ states is probably easier to see); and

(ii) any DFSA that accepts L_n has at least 2^n states.

N.B. This illustrates the power of nondeterminism: For some languages, such as L_n, there can be an exponential amount of savings in the size of the automaton if we use nondeterministic automata. It also shows that the subset construction is, in a sense, optimal: For some languages, such as L_n, no matter how we transform a nondeterministic FSA that accepts the language to a deterministic one, the size of the resulting automaton *has to* increase exponentially. Another example that illustrates the same points is shown in Exercise 12 of Chapter 7 in the notes.