CS 181 Spring 2018 Homework Week 2

Assigned 4/9; Due Mon 4/16 @ 3:00pm at Box A1 CS181 Room BH 2432

From last week:

Two: Let L_a and L_b be the same sets as in the previous homework (as clarified in the Course Announcements and slightly rephrased for clarity here). Let $\Sigma = \{ 0, 1 \}$.

L_a = All strings over Σ that change from one symbol to the other at most once as you read the string

L_b = All strings over Σ that begin and end with the same symbol

d. Note that we can also view Σ as a language consisting of just the two strings 0 & 1. List the elements of the language concatenation:

$$\Sigma \bullet (L_b \cap \{\text{strings of length} \leq 2\})$$

e. What is the language concatenation $L_a \bullet \{\}$?

Six: Consider the two DFAs M_1 & M_2 (Fig. 1.6 on p. 36 & Fig. 1.8 on p. 37, respectively of Sipser) and reproduced here for convenience. What is the language concatenation $\mathcal{L}(M_1) \bullet \mathcal{L}(M_2)$? I.e., give a simple English description of this language.

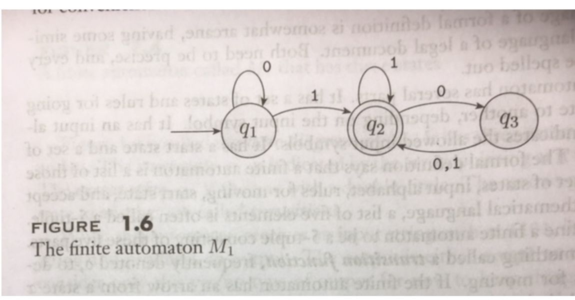

FIGURE 1.6
The finite automaton M_1

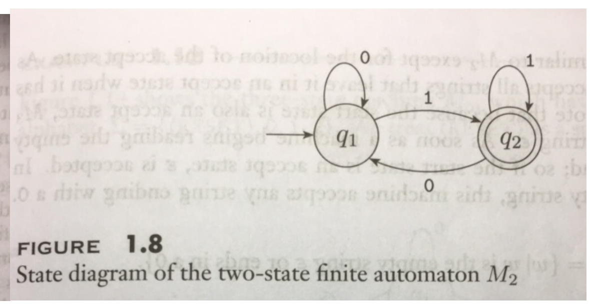

FIGURE 1.8
State diagram of the two-state finite automaton M_2

0. Consider the following alternative definition of the transitive closure δ^* of the transition function δ for a DFA:

For a DFA M = (Q, Σ, δ, q_0, F), extend δ over symbols in Σ to δ^* over strings in Σ^* thusly:

Define $\delta^* : (Q \times \Sigma^*) \rightarrow Q$:

$$\delta^*(q, \varepsilon) = q \text{ , for all } q \in Q$$

$$\delta^*(q, xb) = \delta(\delta^*(q, x) , b), \text{ for all } q \in Q, b \in \Sigma , \& \ x \in \Sigma^*$$

a. How would you interpret this version of the definition in English vs. the way we interpreted the usual definition?

b. Prove this version is equivalent to the usual definition by showing that for any DFA, M, it gives the same resulting state for all q in Q and input string w in Σ^*. Since they are both recursive definitions, you'll want to use induction.

1. Let M = (Q, Σ, δ, q_0, F) be a DFA with Σ = {a, b}, Q = {(0,0), (0,1), (1,0), (1,1)}, q_0 = (0,0), F = {q_0}, and δ given by the formulas:

   δ((x, y), a) = ((x+1) MOD 2 , y)

   δ((x, y), b) = (x , (y+1) MOD 2)

   where x, y $\in$ { 0, 1 }.

   a. Give the state transition table for transition function δ.

   b. Draw the state diagram for DFA M. (Don't forget the initial state and final state(s).)

   c. Describe in English the language recognized by this DFA.

   d. How would you interpret the names of the states in Q in English in terms of how the DFA works?

Inspired by Sipser Exercises: pp. 83-86:

2. Show a DFA over the alphabet { a, b } for the following language (from Sipser Exercise 1.5.h) via the complement construction (as in Exercise 1.14.a):

   { w $\in \Sigma^*$ | w is not the string a nor the string b }

3. Show an NFA over Σ = {0, 1} for the language in Sipser 1.6l:

   L_{three} = { w $\in \Sigma^*$ | w contains an even number of 0's or it contains exactly two 1's}

   Can you do it with 6 states?

4. For the following language over Σ = {a, b} (from Sipser Exercise 1.20.h), give two strings which are in the language and two which are not (total of four strings).

   $(a \cup ba \cup bb)\ \Sigma^*$

5. Consider the following two languages over Σ = { a, b, c }.

   L_{five} = { xby | x, y $\in$ { a, c }* and the number of a's in x = the number of a's in y }

   L_{NFS} = { $a^n b a^n$ | $n \geq 0$ }

   As discussed in class, languages like L_{NFS} are *not* finite state. Prove that L_{five} is not finite state using closure properties of finite state languages and the fact that L_{NFS} is not finite state.

eof