CS 581 Assignment 1

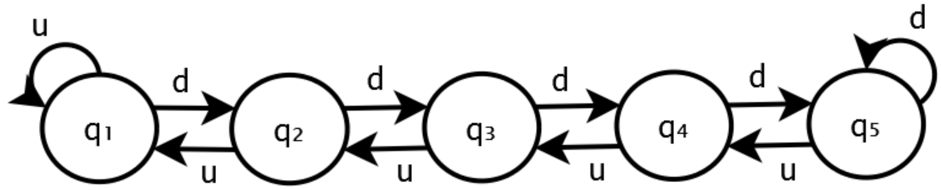
Jessica Fortier

October 11, 2011

1.3 $M = (\{q_1, q_2, q_3, q_4, q_5\}, \{u, d\}, \delta, q_3, \{q_3\})$

$\delta =$

	u	d
q_1	q_1	q_2
q_2	q_1	q_3
q_3	q_2	q_4
q_4	q_3	q_5
q_5	q_4	q_5



1.4(g) $\Sigma = a, b$

$K = \{w|w$ has even length and an odd number of a's$\} = K_1 \cap K_2$ for two simpler languages, K_1 and K_2. Construct DFA's for K_1 and K_2, then combine them to give the state diagram of a DFA for the language given.

Let $K_1 = \{x|x$ has an even length$\}$ and is represented by a DFA $M_1 = (\{s_0, s_1\}, \Sigma, \delta_1, s_0, \{s_0\})$ where:

$\delta_1 =$

	a	b
s_0	s_1	s_1
s_1	s_0	s_0

Let $K_2 = \{y|y$ has an odd number of a's$\}$ and is shown by DFA $M_1 = (\{s_0, s_1\}, \Sigma, \delta_2, s_0, \{s_1\})$ where:

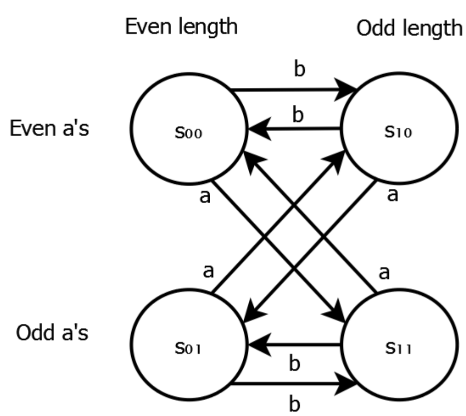
$\delta_2 =$

	a	b
s_0	s_1	s_0
s_1	s_0	s_1

By combining these two DFAs, we can then define a DFA for K called $M = (\{s_{00}, s_{01}, s_{10}, s_{11}\}, \Sigma, \delta, s_{00}, \{s_{01}\})$.

$\delta =$

	a	b
s_{00}	s_{11}	s_{10}
s_{01}	s_{10}	s_{11}
s_{10}	s_{01}	s_{00}
s_{11}	s_{00}	s_{01}



1.5(d) $\Sigma = a, b$

$L = \{w | w \text{ is any string not in } a^*b^*\} = \overline{B}$ for a simpler language B. Construct a DFA for B, then use it to give the state diagram of a DFA for L.

$\delta_b =$

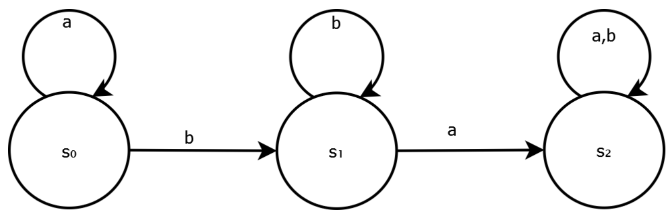
	a	b
s_0	s_0	s_1
s_1	s_2	s_1
s_2	s_2	s_2

$B = \{w | w \text{ is any string in } a^*b^*\}$ and can be represented by a DFA $M = (\{s_0, s_1, s_2\}, \Sigma, \delta_b, s_0, \{s_0, s_1\})$. The DFA representative of $\overline{B}$ (equivalent to L) is that which performs the same identification processes as M, but which accepts where M does not.

Thus, L is represented by DFA $K = (\{s_0, s_1, s_2\}, \Sigma, \delta, s_0, \{s_2\})$. The state diagram for this DFA is given below.

$\delta =$

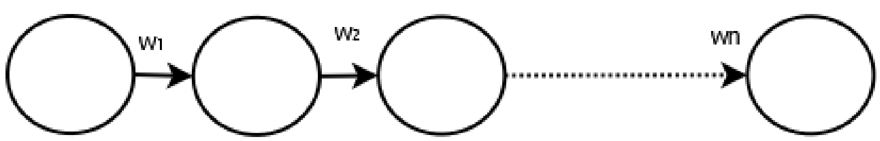
	a	b
s_0	s_0	s_1
s_1	s_2	s_1
s_2	s_2	s_2



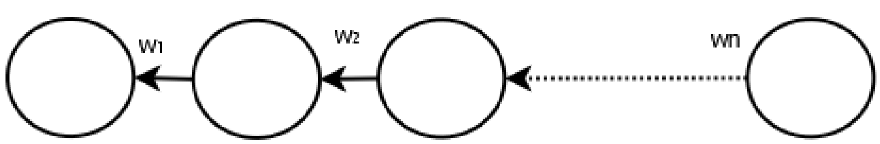
1.31 $A = \{w_1 w_2 ... w_n | w \epsilon A\}$ and $A^{\Re} = \{w^{\Re} | w^{\Re} = w_n ... w_2 w_1 \wedge w \epsilon A\}$

Show that if language A is regular, so is $A^{\Re}$.

In order for A to be regular, it must be recognizable by a finite automaton. This means that for every member of A, a path through such a finite automaton from start state to some accept state must exist.



If one views these accept states for w_n instead as start states, the original start state as an end state, and the direction of state transitions on these same paths as reversed, then it can be said that the path instead is detecting $w^{\Re}$.



A finite automaton can have only one start state by definition, so consider that a single state is added to the automaton and possesses epsilon transitions to each state which formerly accepted on a w_n value (and now originates the detection of a $w^{\Re}$ string). We have designed this automaton to recognize all strings $w^{\Re}$, which is the set of strings belonging to language $A^{\Re}$. By definition, this language must also be regular because a finite automaton does exist to recognize it.

1.32 $\Sigma_3 = \left\{ \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \ldots, \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \right\}$

$B = \{w \mathrel{\epsilon} \Sigma_3^* \mid$ the bottom row of w is the sum of the top two rows$\}$. Show that B is regular.

If B is regular, then so is $B^{\Re}$, and vice versa. If a finite automaton can be constructed to recognize $B^{\Re}$, then $B^{\Re}$. This implies that B regular as well.

A DFA describing $B^{\Re}$ is $M = (\{nc, ca, in\}, \Sigma_3, \delta, nc, \{nc\})$

$$\delta = \begin{array}{r|cccccccc} & \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} & \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} & \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} & \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} & \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} & \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} & \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \\ \hline \text{(no carry) } nc & nc & in & in & nc & in & nc & ca & in \\ \text{(carry) } ca & in & nc & ca & in & ca & in & in & ca \\ \text{(invalid) } in & in & in & in & in & in & in & in & in \end{array}$$

This automaton tracks the carries and sum bits as the three bit strings are read in from least significant bit to most significant bit. Whenever a carry is generated, the bit strings become unacceptable. Once the carry is assimilated into the sum by an addition with a zero, the bit strings once again become acceptable. There are bit strings which do not satisfy rules of bitwise addition, and these bit strings are also unacceptable.

Because this automaton suffices to recognize $B^{\Re}$, it follows that $B^{\Re}$ is a regular language. By the reversal rules from the previous problem, this requires that the language B is also regular.