CS 4510 Automata and Complexity 8/25/2022

Homework 1: Finite Automata

Due: 9/6/2022

This assignment is due on **11:59 PM EST, Thursday, Sep 6, 2022**. You may turn it in up to 48 hours late, but assignments turned in by the deadline receive 3% extra credit. Additionally, late submission means late feedback, which means less time to study before an exam.

You should submit a typeset or *neatly* written pdf on Gradescope. The grading TA should not have to struggle to read what you've written; if your handwriting is hard to decipher, you will be asked to typeset your future assignments.

You may collaborate with other students, but any written work should be your own.

0. **Background Knowledge** (0 points)

   You should be familiar with concepts such as sets and set operations, functions, graphs, and proofs by contradiction and induction. The following problems in the book can help you review them. (Do not turn these in! We will not grade them.)

   0.2 b,e,f

   0.3 e,f

   0.5

   0.6

   0.8

   0.11

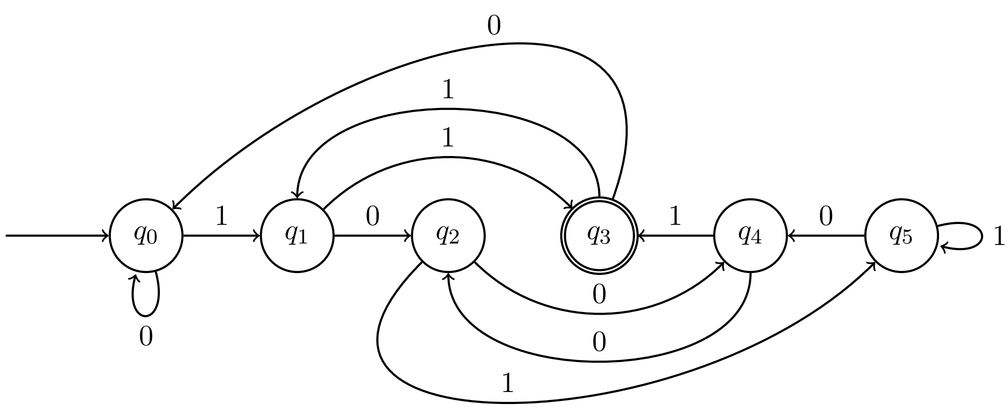
1. **Explicit Construction** ***Section A only***

Give the state diagrams for the DFA's that accept the following languages:

a) (4 points) $\{x | x$ is a binary string, which, when interpreted as a binary number, is equivalent to 3 *mod* 6$\}$

Solution:

input / $\bar{w}$ mod 6	0	1
0 (state q_0)	0	1
1 (state q_1)	2	3
2 (state q_2)	4	5
3 (state q_3)	0	1
4 (state q_4)	2	3
5 (state q_5)	4	5



b) (4 points) $\{x|x$ is a binary string which contains an even number of 1's and ends in 010$\}$ (Hint: language operations)

Solution:

To construct a DFA that accepts a binary string that contains an even number of 1's and ends in 010, we can construct 2 DFAs:

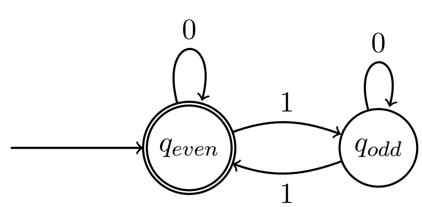
- one DFA accepts a binary string containing an even number of 1's
- other DFA accepts a binary string ending in 010

We can take the 2 DFAs and take the cartesian product which results in the DFA that accepts a binary string containing an even number of 1's and ends in 010.

—

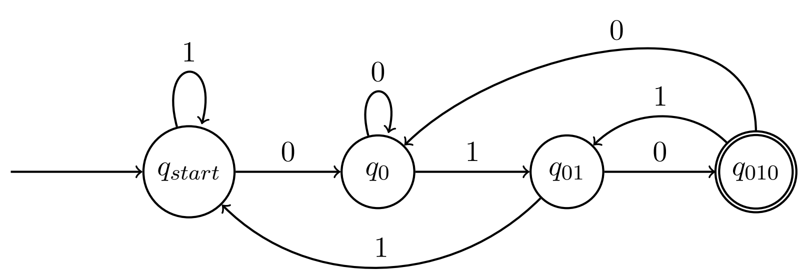
DFA that accepts a binary string which contains an even number of 1's:

states \ input	0	1
0 (state q_{even})	0	1
1 (state q_{odd})	1	0



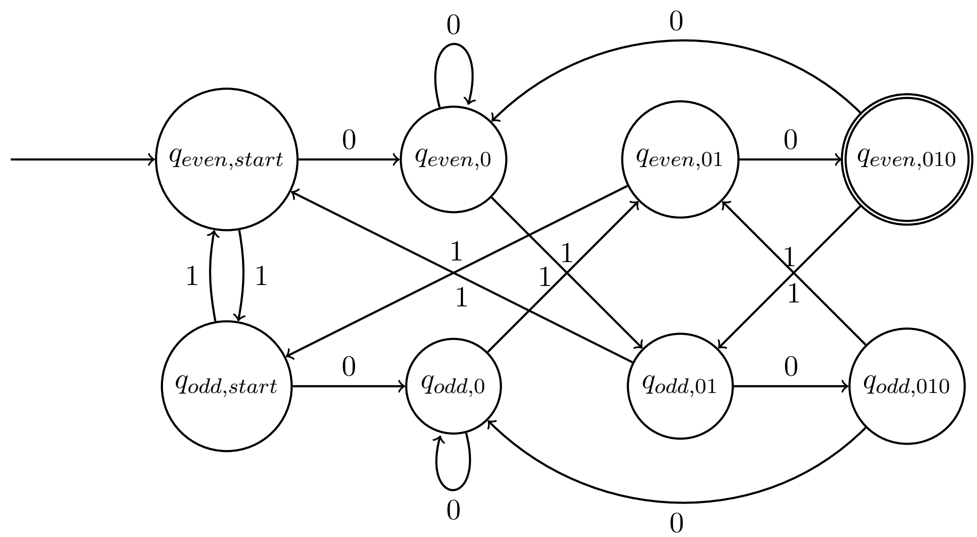
DFA that accepts a binary string which ends in 010:

states \ input	0	1
0 (state q_{start})	1	0
1 (state q_0)	1	2
2 (state q_{01})	3	0
3 (state q_{010})	1	2



Taking the cartesian product, we get the following DFA:

states \ input	0	1
0 (state $q_{even,start}$)	1	4
1 (state $q_{even,0}$)	1	6
2 (state $q_{even,01}$)	3	4
3 (state $q_{even,010}$)	1	6
4 (state $q_{odd,start}$)	5	0
5 (state $q_{odd,0}$)	5	2
6 (state $q_{odd,01}$)	7	0
7 (state $q_{odd,010}$)	5	2



2. **Closure** (4 points)

Let $ONE{-}OR{-}ALL(L_1, L_2, L_3)$ be the set which contains all strings which are either members of all three languages L_1, L_2, L_3 or are contained in *exactly* one of the three languages. Show that, if L_1, L_2, and L_3 are regular, then $ONE{-}OR{-}ALL(L_1, L_2, L_3)$ is also regular.

Solution:

Proof. We know that regular languages are closed under union, intersection, and complement set operations.

Using this fact, we can define the set which contains exactly one of L_1, L_2, L_3 as $(L_1 \cap \bar{L_2} \cap \bar{L_3}) \cup (\bar{L_1} \cap L_2 \cap \bar{L_3}) \cup (\bar{L_1} \cap \bar{L_2} \cap L_3)$ (strings in one language but not other languages is simply the intersection of that language with the complement of all the other languages).

We can define the set which contains all strings in all of L_1, L_2, L_3 as $(L_1 \cap L_2 \cap L_3)$.

Then we can combine the two definitions together to form the set $ONE - OR - ALL(L_1, L_2, L_3) = (L_1 \cap \bar{L_2} \cap \bar{L_3}) \cup (\bar{L_1} \cap L_2 \cap \bar{L_3}) \cup (\bar{L_1} \cap \bar{L_2} \cap L_3) \cup (L_1 \cap L_2 \cap L_3)$.

Because the expression created is formed through the union, intersection, and complement set operations, $ONE - OR - ALL(L_1, L_2, L_3)$ is also regular as L_1, L_2, L_3 are regular languages and will stay closed under the expression.

□

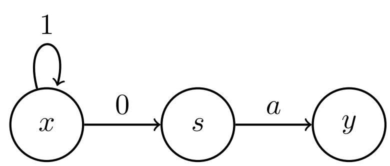
3. **More Closure** (4 points)

   Let $L[0]$ be defined formally as $\{x_1 0 x_2 ... 0 x_n | x_i \in \Sigma, x_1 ... x_n \in L\}$. In other words, strings of $L[0]$ are formed by taking a string from L and inserting a 0 between each character of the word, so that if $L = \{11, 1010\}$, then $L[0] = \{101, 1000100\}$. Prove that if L is regular, so is $L[0]$.

Solution: By definition, a language L is regular if there is a finite automata accepting it. If L is regular, let a finite automata M_1 accept it, where $M_1 = (Q_1, \Sigma, \delta^1, q_0^1, F_1)$.

As for $L[0]$, we can construct a finite automata M_2 that accepts $L[0]$ with the following:

- $Q_2 = Q_1$
- $\Sigma = \Sigma$
- For δ^2 between each transition, we want to add an intermediate state. So, for some state x that transitions to state y on action $a \in \delta_1$ and $x, y \in Q_1$, we add an intermediate state s such that x loops back to itself on 1 and moves to the intermediate state s on 0. And s only transitions to state y on action a. Below, I show the transitions:



- $q_0^2 = q_0^1$
- $F_2 = \{x_1 0 x_2 0 ... 0 x_n | x_i \in \Sigma, x_1 ... x_n \in F_1\}$

Because we can construct a finite automata M_2 which accepts $L[0]$ using a combination of the finite automata M_1, by the definition above $L[0]$ is also regular (if L is also regular).

4. **Even More Closure** ***Section X only*** (4 points)

   Let $END(L) = \{x \mid \text{ for some } y \in \Sigma^*,\ yx \in L\}$. Thus $END(L)$ is the set of strings which could appear as a suffix of a string in L. Prove that if L is regular, $END(L)$ is regular.

5. **Minimal DFA's** ***Section X only*** (4 points)

   Let L_k be the regular language over $\{0, 1\}$ which contains all strings which have a 1 as the k'th character from the right end of the string. Prove that any DFA that recognizes L_k must contain at least 2^k states.