CS 3133: Homework 3

Adam Camilli (aocamilli@wpi.edu)

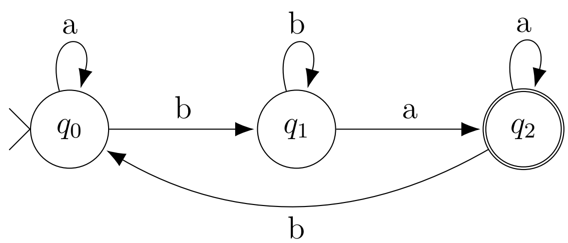
September 18, 2017

1. **5.1** (184) Let M be the DFA defined by

$$Q = \{q_0, q_1, q_2\}$$
$$\sum = \{a, b\}$$
$$F = \{q_2\}$$

δ	a	b
q_0	q_0	q_1
q_1	q_2	q_1
q_2	q_2	q_0

(a) Give the state diagram of M.



(b) Trace the computations of M that process the strings *abaa*, *bbbabb*, *bababa*, and *bbbaa*.

$$\begin{aligned} & [q_0, abaa] \\ \vdash\ & [q_0, baa] \\ \vdash\ & [q_1, aa] \\ \vdash\ & [q_2, a] \\ \vdash\ & [q_2, \lambda] \ \checkmark \text{ (Accept)} \end{aligned}$$

$$\begin{aligned} & [q_0, bbbabb] \\ \vdash\ & [q_1, bbabb] \\ \vdash\ & [q_1, babb] \\ \vdash\ & [q_1, abb] \\ \vdash\ & [q_2, bb] \\ \vdash\ & [q_0, b] \\ \vdash\ & [q_1, b] \\ \vdash\ & [q_1, \lambda] \ \mathbf{X} \text{ (Reject)} \end{aligned}$$

$$\begin{aligned} & [q_0, bababa] \\ \vdash\ & [q_1, ababa] \\ \vdash\ & [q_2, baba] \\ \vdash\ & [q_0, aba] \\ \vdash\ & [q_0, ba] \\ \vdash\ & [q_1, a] \\ \vdash\ & [q_2, \lambda] \ \checkmark \text{ (Accept)} \end{aligned}$$

$$\begin{aligned} & [q_0, bbbaa] \\ \vdash\ & [q_1, bbaa] \\ \vdash\ & [q_1, baa] \\ \vdash\ & [q_1, aa] \\ \vdash\ & [q_2, a] \\ \vdash\ & [q_2, \lambda] \ \checkmark \text{ (Accept)} \end{aligned}$$

(c) Which of the strings from part (b) are accepted by M?

All of them except *bbbabb*.

(d) Give a regular expression for L(M).

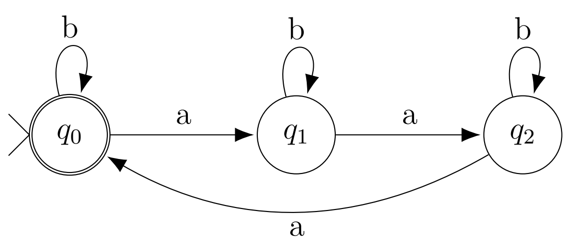
$$a^\star b^+ a^+ (b a^\star b^+ a^+)^\star$$

2. **5.11** (185) Build a DFA that accepts the set of strings over $\{a, b\}$ in which the number of a's is divisible by three.

This set of strings is equivalent to the regular expression $(b^\star a b^\star a b^\star a b^\star)^\star$ which can be modeled in the following state diagram where each state $q_i \in Q$ represents the remainder of current number of a's divided by three (derived from previous state).

q_i	a	b
q_0	q_1	q_0
q_1	q_2	q_1
q_2	q_0	q_2

which corresponds to the state diagram

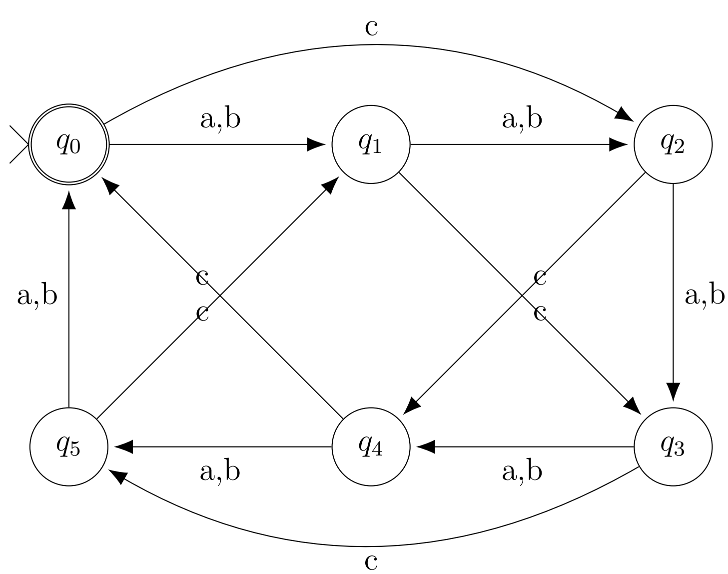


3. Design a DFA that accepts the language consisting of the set of those strings over $\{a, b, c\}$ in which the number of a's plus the number of b's plus twice the number of c's is divisible by six.

The language L that we wish to accept is $L = \{a^l \cup b^m \cup c^n \mid (l+m+2n) \bmod 6 = 0\}$. Designing a regular expression for this language is quite difficult, so it will be more efficient to jump straight to a state table to describe the behavior of a DFA M that accepts this language, with each state $q_i \in Q$ representing the remainder i of dividing the current $(l + m + 2n)$ by 6 (derived from previous state). Since a and b affect this value equally, they are grouped together:

q_i	(a, b)	c
q_0	q_1	q_2
q_1	q_2	q_3
q_2	q_3	q_4
q_3	q_4	q_5
q_4	q_5	q_0
q_5	q_0	q_1

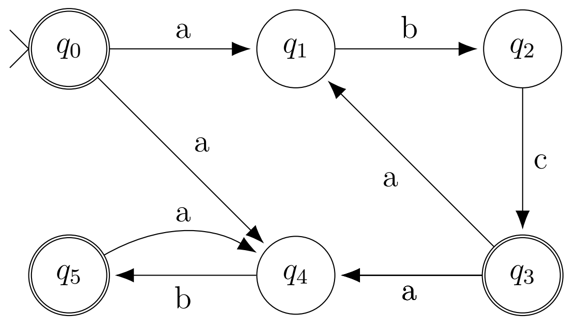
From here, we can draw the state diagram:



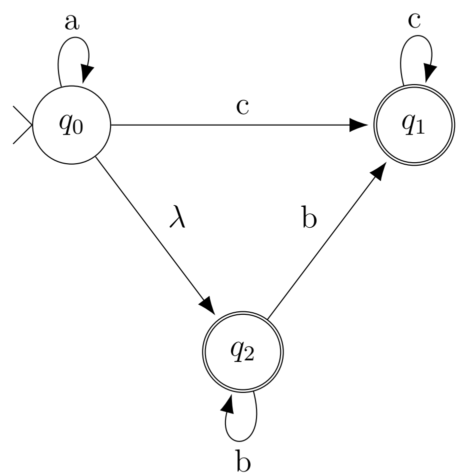
4. Draw an NFA that accepts the following language over the alphabet a, b, c:

$$\boldsymbol{(abc)^\star(ab)^\star}$$

Since this language includes λ, we make the starting state accepting, and simply ensure that from there $\nexists$ any path to an accepting state that does not go from a to b or from a to b to c. Since this is an NFA, we are perfectly fine with only assigning transitions to "correct" inputs and allowing "incorrect" inputs such as ac or b to cause the automaton to choke and not progress.



5. **5.36** (187) Let M be the NFA-λ



(a) Compute λ-$closure(q_i)$ for $i = 0, 1, 2$.

$$\begin{aligned}\lambda\text{-}(q_0) &= \{q_0, q_2\}\\ \lambda\text{-}(q_1) &= \{q_1\}\\ \lambda\text{-}(q_2) &= \{q_2\}\end{aligned}$$

(b) Give the input transition function t for M.

Defining

$$t(q_i, a) = \bigcup_{q_j \in \lambda\text{-}closure(q_i)} \lambda\text{-}closure(\delta(q_j, a))$$

we may convert from

δ	a	b	c	λ
q_0	$\{q_0\}$	$\emptyset$	$\{q_1\}$	$\{q_2\}$
q_1	$\emptyset$	$\emptyset$	$\{q_1\}$	$\emptyset$
q_2	$\emptyset$	$\{q_2, q_1\}$	$\emptyset$	$\emptyset$

$\Rightarrow$

t	a	b	c
q_0	$\{q_0, q_2\}$	$\{q_2, q_1\}$	$\{q_1\}$
q_1	$\emptyset$	$\emptyset$	$\{q_1\}$
q_2	$\emptyset$	$\{q_2, q_1\}$	$\emptyset$

(c) Use Algorithm 5.6.3 to construct a state diagram of a DFA that is equivalent to M.

i. Initialize a new set of states Q′ as λ-$closure(q_0)$.

$$Q' = \{q_0, q_2\}$$

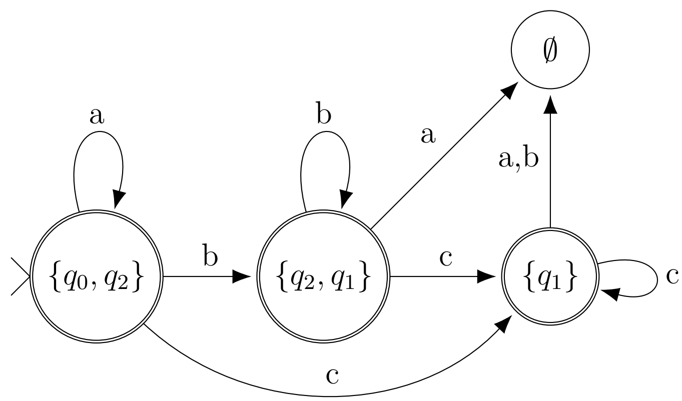
ii. Begin a transition table with $\sum\{a, b, c\}$ using Q′:

δ	a	b	c
$\{q_0, q_2\}$	q_0	$\{q_2, q_1\}$	q_1

iii. Here we see two new states $\{q_1\}, \{q_2, q_1\} \notin$ Q′, so add them to the table:

δ	a	b	c
$\{q_0, q_2\}$	$\{q_0\}$	$\{q_2, q_1\}$	$\{q_1\}$
$\{q_2, q_1\}$	$\emptyset$	$\{q_2, q_1\}$	$\{q_1\}$
$\{q_1\}$	$\emptyset$	$\emptyset$	$\{q_1\}$

iv. We have no more new states, so now we may draw a DFA, with all invalid inputs going to a "death state":



(d) Give a regular expression for L(M).

$$\text{L(M)} = a^\star b^\star c^\star$$