Recit-4

November 2, 2020

Recall: If L and M are regular languages, then $L \cap M$ is also a regular language.

This can be proved by finding a DFA that accepts $L \cap M$. Let DFA A_L accepts L and DFA A_M then, the DFA that accepts $L \cap M$ is the product of two automaton $A_L \times A_M$.

$$L(A_M \times A_N) = L(A_M) \cap L(A_N) \tag{1}$$

where L(A) is the language accepted by A.

Q1

Product Automaton and Intersection of Languages

Let $\Sigma = \{0, 1\}$;

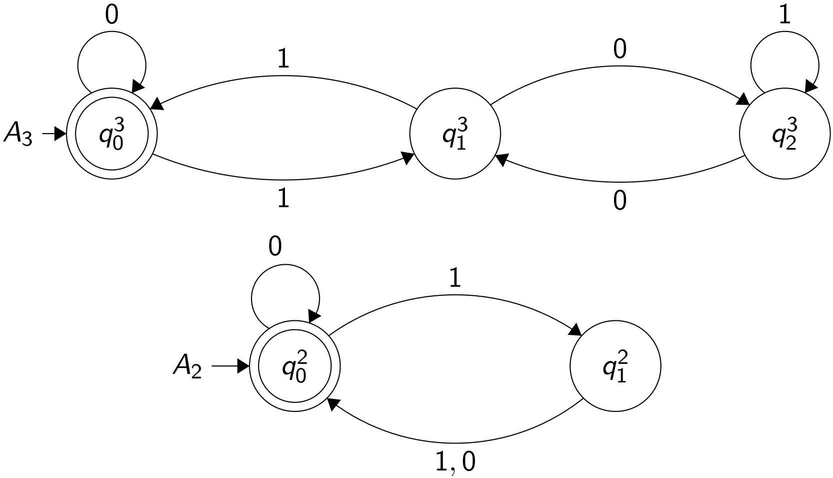
Define $L(n) = \{u \in (0+1)^* \mid n | binary(u)\}$ where binary(u) is the binary evaluation of string u and $binary(\epsilon) = 0$.

Question) What is the DFA that accepts $L(6)$?

- Any number that is divisible by 2 **AND** 3 is also divisible by 6.
- So we can say that $L(6) = L(3) \cap L(2)$.
- Let A_n accepts $L(n)$, then A_6 is equivalent to $A_3 \times A_2$.

Q1

Product Automaton and Intersection of Languages



$A_6 = A_3 \times A_2;$

$Q_{A_6} = Q_{A_3} \times Q_{A_2};$

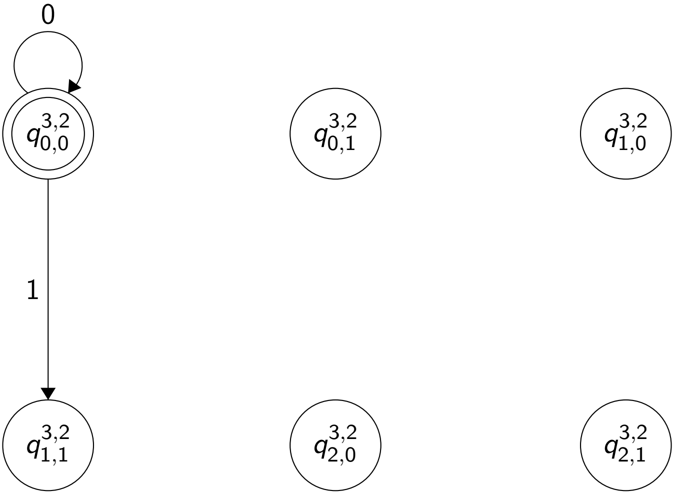
$F_{A_6} = F_{A_3} \times F_{A_2};$

$Q_{0_{A_6}} = Q_{0_{A_3}} \times Q_{0_{A_2}};$

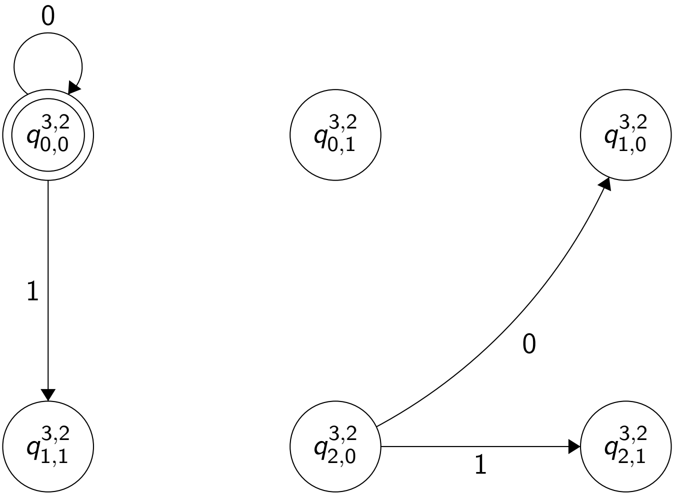
$\delta_{A_6}((q, r), u) = (\delta_{A_3}(q, u), \delta_{A_2}(r, u));$

- $Q_{A_6} = \{(q_0^3, q_0^2), (q_1^3, q_0^2), (q_1^3, q_1^2), (q_1^3, q_1^2), (q_2^3, q_1^2), (q_2^3, q_1^2)\}$
- $F_{A_6} = \{(q_0^3, q_0^2)\}$
- $Q_{0_{A_6}} = \{(q_0^3, q_0^2)\}$

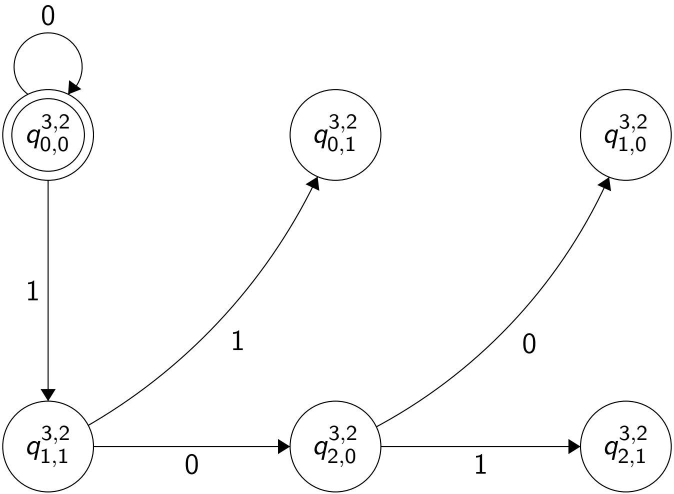
Q1



$\delta_{A_6}((q_0, q_0), 0) = (\delta_{A_3}(q_0, 0), \delta_{A_2}(q_0, 0)) = (q_0, q_0)$
$\delta_{A_6}((q_0, q_0), 1) = (\delta_{A_3}(q_0, 1), \delta_{A_2}(q_0, 1)) = (q_1, q_1)$



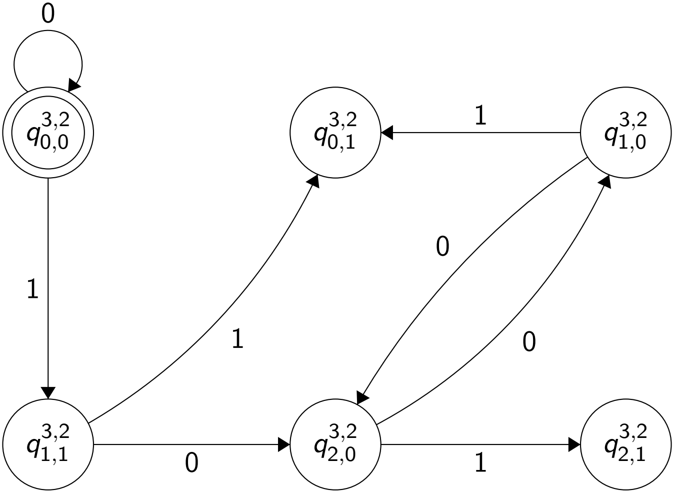
$$\delta_{A_6}((q_2, q_0), 0) = (\delta_{A_3}(q_2, 0), \delta_{A_2}(q_0, 0)) = (q_1, q_0)$$
$$\delta_{A_6}((q_2, q_0), 1) = (\delta_{A_3}(q_2, 1), \delta_{A_2}(q_0, 1)) = (q_2, q_1)$$



$\delta_{A_6}((q_1, q_1), 0) = (\delta_{A_3}(q_1, 0), \delta_{A_2}(q_1, 0)) = (q_0, q_1)$
$\delta_{A_6}((q_1, q_1), 1) = (\delta_{A_3}(q_1, 1), \delta_{A_2}(q_1, 1)) = (q_2, q_0)$

Q1

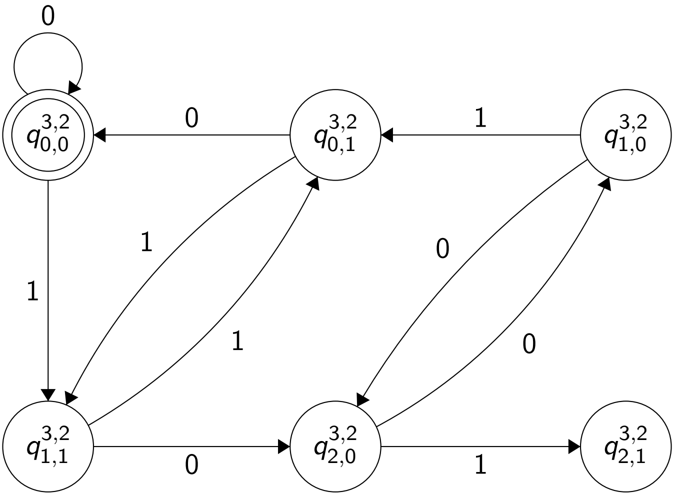
Product Automaton and Intersection of Languages



$$\delta_{A_6}((q_1, q_0), 0) = (\delta_{A_3}(q_1, 0), \delta_{A_2}(q_0, 0)) = (q_2, q_0)$$
$$\delta_{A_6}((q_1, q_0), 1) = (\delta_{A_3}(q_1, 1), \delta_{A_2}(q_0, 1)) = (q_0, q_1)$$

Q1

Product Automaton and Intersection of Languages

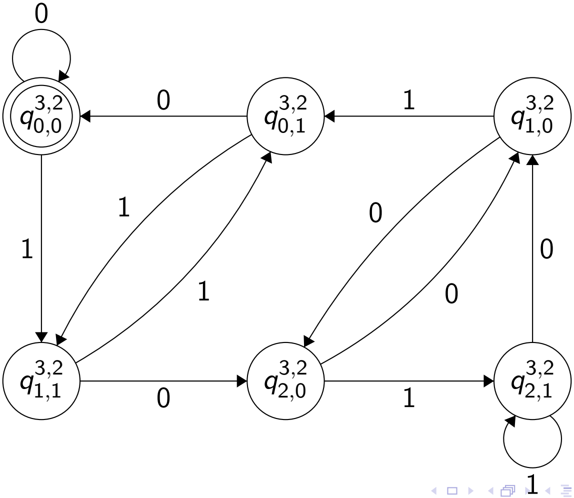


$$\delta_{A_6}((q_0, q_1), 0) = (\delta_{A_3}(q_0, 0), \delta_{A_2}(q_1, 0)) = (q_0, q_0)$$
$$\delta_{A_6}((q_0, q_1), 1) = (\delta_{A_3}(q_0, 1), \delta_{A_2}(q_1, 1)) = (q_1, q_1)$$

Q1

Product Automaton and Intersection of Languages

$$\delta_{A_6}((q_2, q_1), 0) = (\delta_{A_3}(q_2, 0), \delta_{A_2}(q_1, 0)) = (q_1, q_0)$$
$$\delta_{A_6}((q_2, q_1), 1) = (\delta_{A_3}(q_2, 1), \delta_{A_2}(q_1, 1)) = (q_2, q_1)$$



Q2

Reversal of Strings and State Elimination

Let L be the language generated by regular expression $(0.1)^*.1$. Check if L^R is a regular language, if so find the DFA R that accepts L^R. Then find the RE that generates L^R.

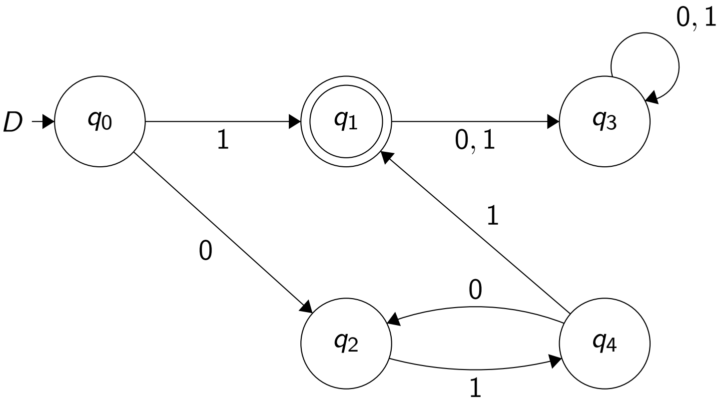
A language L is regular if and only if there exists a DFA that accepts L.

1. Find the DFA D that accepts L.
2. Convert DFA D to the DFA R which accepts L^R.
3. Find the RE that generates L^R by applying state elimination on DFA R.

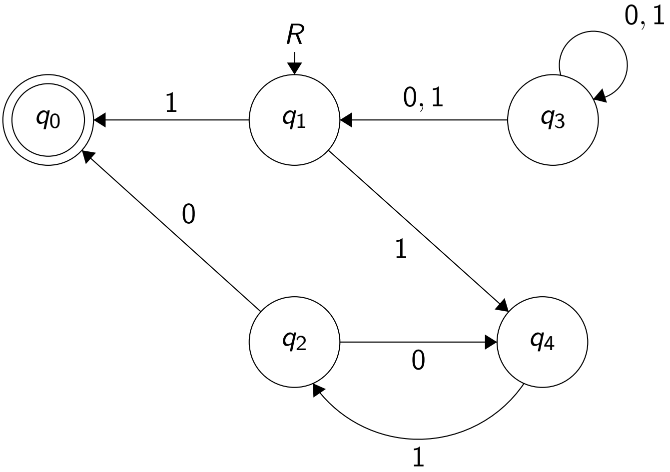
Q2

Reversal of Languages and State Elimination

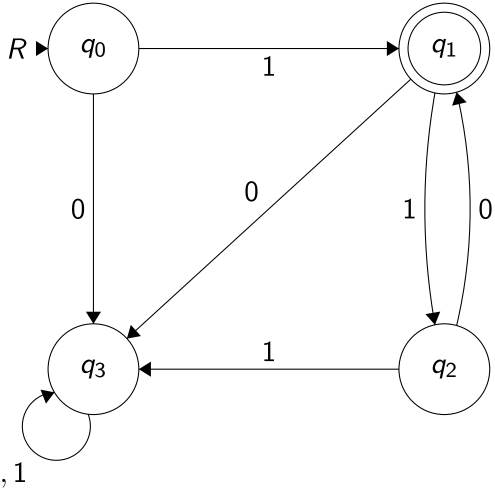
$$(0.1)^*.1 \tag{2}$$



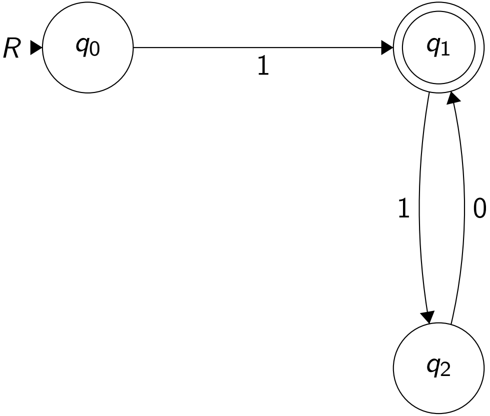
NFA that accepts L^R obtained from the former by (i) interchanging initial and final state sets; and (ii) changing the direction of all transition arcs.



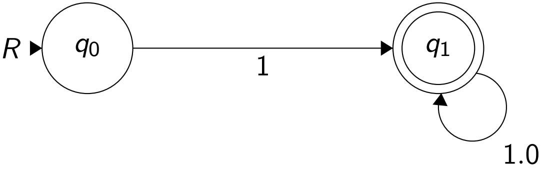
DFA version of R^N;



First eliminate q_3 to make DFA simpler. Because it has no outgoing transition.

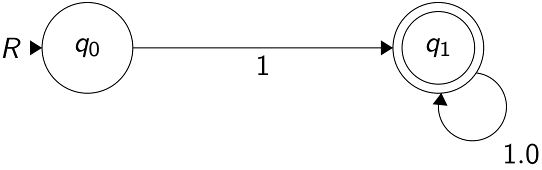


Eliminate q_2;



Q2

Reversal of Languages and State Elimination



$L^R := 1.(1.0)^*$

L^R is a regular language.

If L is a language, and a is a symbol, then L/a , the quotient of L and a , is the set of strings w such that $w.a$ is in L. For example, if $L = \{a, aab, baa\}$, then $L/a = \{\epsilon, ba\}$. Prove that if L is regular, so is L/a.

- We need to find a DFA D_q that accepts L/a to show that it is a regular language, given L is regular and D is the DFA that accepts L.
- Let $D = (Q, \Sigma, \delta, Q_0, F)$, $L(D) = L$, $a \in \Sigma$.

Define the F_a be the set of states $\{q \in Q \mid \delta(q, a) \in F\}$. Note that F_a might be empty.
Then the DFA $D_{q(a)} = (Q, \Sigma, \delta, Q_0, F_a)$ accepts L/a.

The only difference between $D_{q(a)}$ and D is the set of accepting state F and F_a.

If a string $w \in L/a$ then, $w.a \in L$. Because $\delta E(q_0, w) = q$ which implies $q \in F_a$, so $\delta(q, a) = q_f$ where $q_f \in F$.

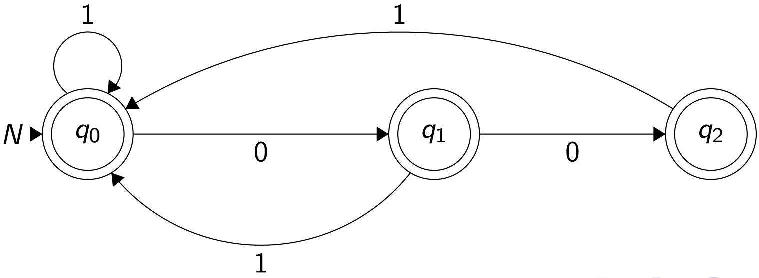
$L(D_{q(a)}) = L/a$.

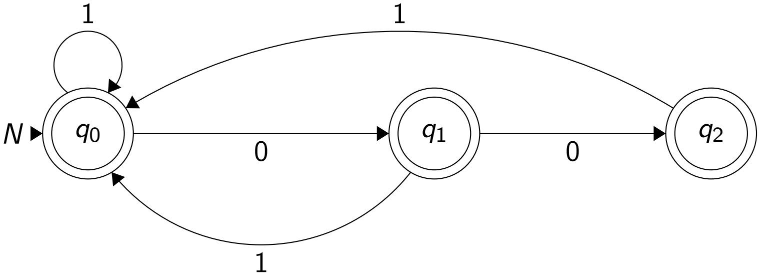
$$(1 + 01 + 001)^*.(\epsilon + 0 + 00) \tag{3}$$

Simplify the above regular expression and describe it in simple natural language.

$$\equiv ((\epsilon + 0 + 00)1)^*(\epsilon + 0 + 00) \tag{4}$$

$$\equiv ((\epsilon + 0)(\epsilon + 0)1)^*(\epsilon + 0)(\epsilon + 0) \tag{5}$$





All binary strings which do not contain three consecutive 0's.

$$0^* + 0^*1(\epsilon + 00^*1)^*000^* \tag{6}$$

Simplify the above regular expression.

$$\equiv 0^* + 0^*1(00^*1)^*000^* \tag{7}$$

By using $\epsilon + \alpha\alpha^* \equiv \alpha^*$ and $(\alpha\beta)^*\alpha \equiv \alpha(\beta\alpha)^*$;

$$\equiv \epsilon + 00^* + 0^*10(0^*10)^*00^* \tag{8}$$

$$\equiv \epsilon + (\epsilon + 0^*10(0^*10)^*)00^* \tag{9}$$

$\epsilon + \alpha.\alpha^* = \alpha^*$

$$\equiv \epsilon + (0^*10)^*00^* \tag{10}$$

By using $\alpha\alpha^* \equiv \alpha^*\alpha$;

$$\equiv \epsilon + (0^*10)^*0^*0 \tag{11}$$

Finally, using $(\alpha^*\beta)^*\alpha^* \equiv (\alpha + \beta)^*$

$$\equiv \epsilon + (0 + 10)^*0 \tag{12}$$