Exercise 3.32.

Let $M = (Q, \Sigma, q_0, A, \delta)$ be an NFA accepting a language L. Assume that there are no transitions to q_0, that A has only one element, q_f, and that there are no transitions from q_f.

a. Let M_1 be obtained from M by adding Λ-transitions from q_0 to every state that is reachable from q_0 in M.
(If p and q are states, q is reachable from p if there is a string $x \in \Sigma^*$ such that $q \in \delta^*(p, x)$.)
Describe (in terms of L) the language accepted by M_1.

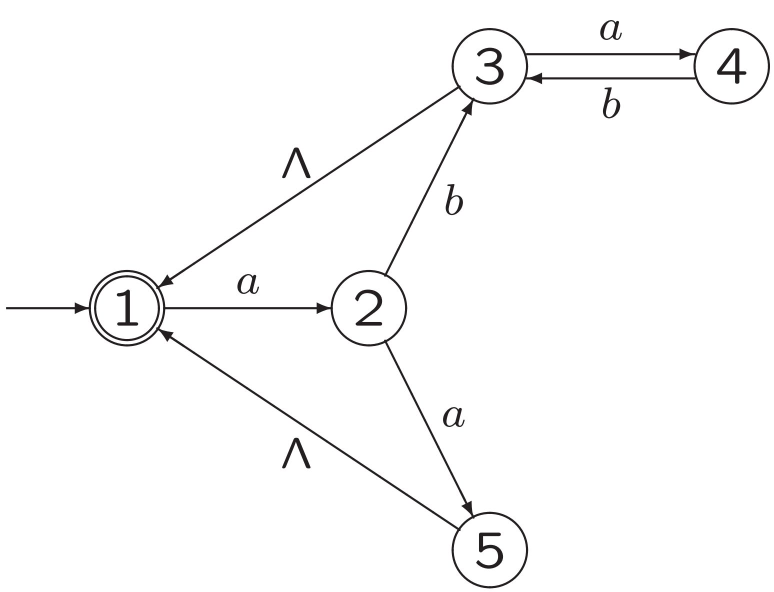
b. Let M_2 be obtained from M by adding Λ-transitions to q_f from every state from which q_f is reachable in M.
Describe (in terms of L) the language accepted by M_2.

c. Let M_3 be obtained from M by adding both the Λ-transitions in (a) and those in (b).
Describe (in terms of L) the language accepted by M_3.

Exercise 3.37.

For each part below, use the algorithm from the lecture to draw an NFA with no Λ-transitions accepting the same language as the NFA pictured.

b.

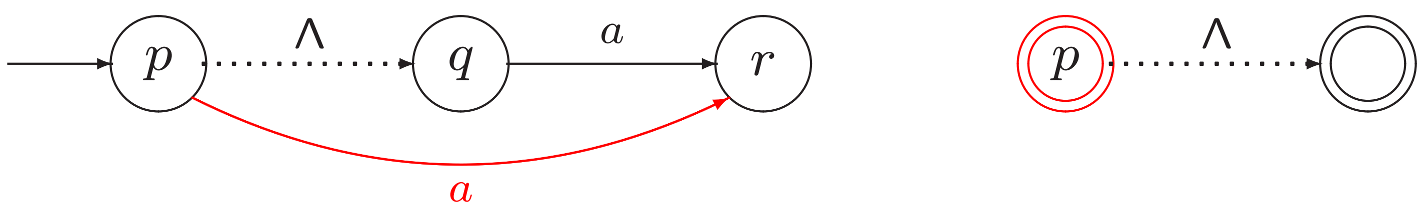


Hint: use the transition table of the NFA, extended with the Λ-closure of every state:

q	$\delta(q,a)$	$\delta(q,b)$	$\delta(q,\Lambda)$	$\Lambda(\{q\})$
1	2	—	—	...
2	5	3	—	...
3	4	—	1	...
4	—	3	—	...
5	—	—	1	...

Exercise.

Our construction:



Λ-removal

Given NFA $M = (Q, \Sigma, \delta, q_0, A)$,

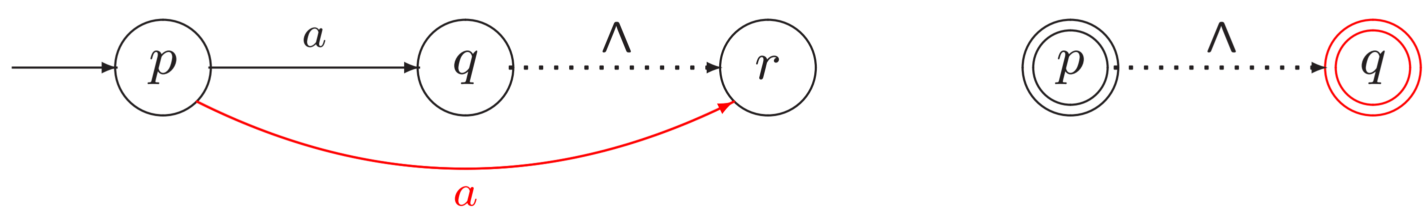
construct NFA $M_1 = (Q, \Sigma, \delta_1, q_0, A_1)$ without Λ-transitions:

- whenever $q \in \Lambda_M(\{p\})$ and $r \in \delta(q, a)$, add r to $\delta_1(p, a)$
- whenever $\Lambda_M(\{p\}) \cap A \neq \emptyset$, add p to A_1.

continued on next slide. . .

Exercise. (ctd.)

Is it possible to invert the construction:



Λ-removal

Given NFA $M = (Q, \Sigma, \delta, q_0, A)$,

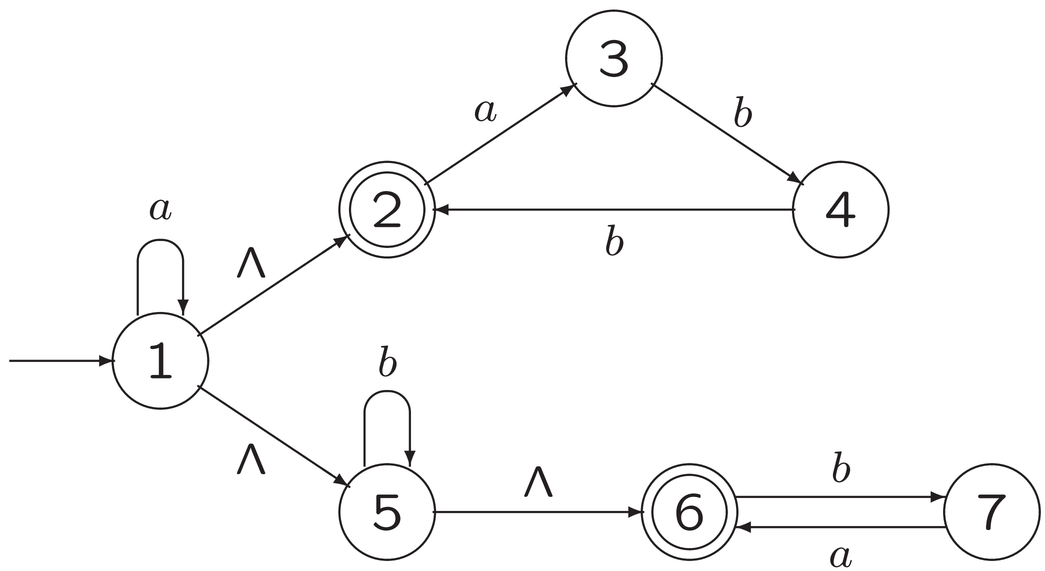
construct NFA $M_1 = (Q, \Sigma, \delta_1, q_0, A_1)$ without Λ-transitions:

- whenever $q \in \delta(p, a)$ and $r \in \Lambda_M(\{q\})$, add r to $\delta_1(p, a)$
- whenever $p \in A$ and $q \in \Lambda_M(\{p\})$, add q to A_1.

Exercise 3.40.

For each part below, draw an FA accepting the same language as the NFA shown.

a.



Hint: First eliminate the Λ-transitions, then apply the subset construction. In both steps, use the transition table to avoid mistakes.

Exercise 3.7.

Find a regular expression corresponding to each of the following subsets of $\{a, b\}^*$.

a. ♣ The language of all strings containing exactly two a's.

c. ♣ The language of all strings that do not end with ab.

e. ♣ The language of all strings not containing the substring aa.

f. ♣ The language of all strings in which the number of a's is even.

Exercise. (Example 3.4)

Find a regular expression corresponding to the language of:

all strings over $\{a, b\}$ in which both the number of a's and the number of b's is even.

Exercise 3.7.

Find a regular expression corresponding to each of the following subsets of $\{a, b\}^*$.

g. ♣ The language of all strings containing no more than one occurrence of the string aa. (The string aaa should be viewed as containing two occurrences of aa.)

i. The language of all strings containing both bb and aba as substrings.

j. The language of all strings not containing the substring aaa.

k. ♣ The language of all strings not containing the substring bba.

l. ♣ The language of all strings containing both aba and bab as substrings.

m. ♣ The language of all strings in which the number of a's is even and the number of b's is odd.

Exercise 3.1. ♣

In each case below, find a string of minimum length in $\{a, b\}^*$ not in the language corresponding to the given regular expression.

a. $b^*(ab)^*a^*$

b. $(a^* + b^*)(a^* + b^*)(a^* + b^*)$

Exercise 3.2. Consider the two regular expressions

$$r = a^* + b^* \qquad s = ab^* + ba^* + b^*a + (a^*b)^*$$

a. Find a string corresponding to r but not to s.
b. Find a string corresponding to s but not to r.
c. Find a string corresponding to both r and s.
d. Find a string in $\{a, b\}^*$ corresponding to neither r nor s.

Exercise 3.10. ♣

a. If L is the language corresponding to the regular expression $(aab + bbaba)^*baba$, find a regular expression corresponding to $L^r = \{x^r \mid x \in L\}$.

b. Using the example in part (a) as a model, give a recursive definition (based on Definition 3.1) of the reverse e^r of a regular expression e.

c. Show that for every regular expression e, if the language L corresponds to e, then L^r corresponds to e^r.