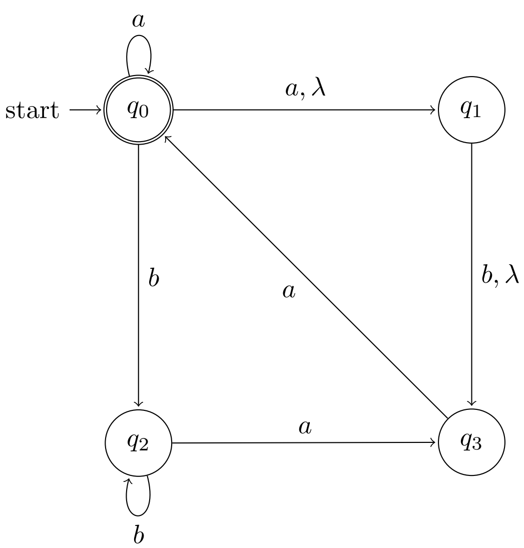
Formal languages, grammars, and automata

Assignment 3, Wednesday, Nov. 26, 2014

Exercises with answers

1. Consider the following NFA$_\lambda$, M.



(a) Compute $\delta^*(q_0, aabaa)$ and $\delta^*(q_0, aabba)$. Are these words accepted?

(b) Which of the following words are accepted by M: $aabbaa$, bbb? Explain.

(c) Compute λ-closure(q_0), λ-closure(q_1), λ-closure(q_2) and λ-closure(q_3).

(d) Eliminate non-determinism and λ-steps (using the subset construction) and compute the associated DFA M'.

Answer Info: . . . .

(a)

$$\begin{aligned}
\delta^*(q_0, aabaa) &= \bigcup\{\delta^*(q', abaa) \mid q' \in \delta(q, a), q \in \lambda\text{-}closure(q_0)\}\\
&= \bigcup\{\delta^*(q', abaa) \mid q' \in \delta(q, a), q \in \{q_0, q_1, q_3\}\}\\
&= \bigcup\{\delta^*(q', abaa) \mid q' \in \{q_0, q_1\}\}\\
&= \bigcup\{\delta^*(q', baa) \mid q' \in \delta(q, a), q \in \lambda\text{-}closure(\{q_0, q_1\})\}\\
&= \bigcup\{\delta^*(q', baa) \mid q' \in \{q_0, q_1\}\}\\
&= \bigcup\{\delta^*(q', aa) \mid q' \in \delta(q, b), q \in \lambda\text{-}closure(\{q_0, q_1\})\}\\
&= \bigcup\{\delta^*(q', aa) \mid q' \in \{q_2, q_3\}\}\\
&= \bigcup\{\delta^*(q', a) \mid q' \in \delta(q, a), q \in \lambda\text{-}closure(\{q_2, q_3\})\}\\
&= \bigcup\{\delta^*(q', a) \mid q' \in \{q_0, q_3\}\}\\
&= \bigcup\{\delta^*(q', \lambda) \mid q' \in \delta(q, a), q \in \lambda\text{-}closure(\{q_0, q_3\})\}\\
&= \bigcup\{\delta^*(q', \lambda) \mid q' \in \{q_0, q_1\}\}\\
&= \{q_0, q_1, q_3\}
\end{aligned}$$

So, aabaa is accepted.
Using the notation of the lecture, we get

$(q_0, aabaa) \vdash (\{q_0, q_1\}, abaa)$

NB. writing $\{q_0, q_1, q_3\}$ is not according to the definition of δ^*, though it gives the right answer in the end.
[An alernative definition of δ^* would be possible]

$\vdash (\{q_0, q_1\}, baa)$
$\vdash (\{q_3, q_2\}, aa)$
$\vdash (\{q_0, q_3\}, a)$
$\vdash (\{q_0, q_1\}, \lambda)$
$\vdash \{q_0, q_1, q_3\}$

For aabba we get (with some simplified notation):

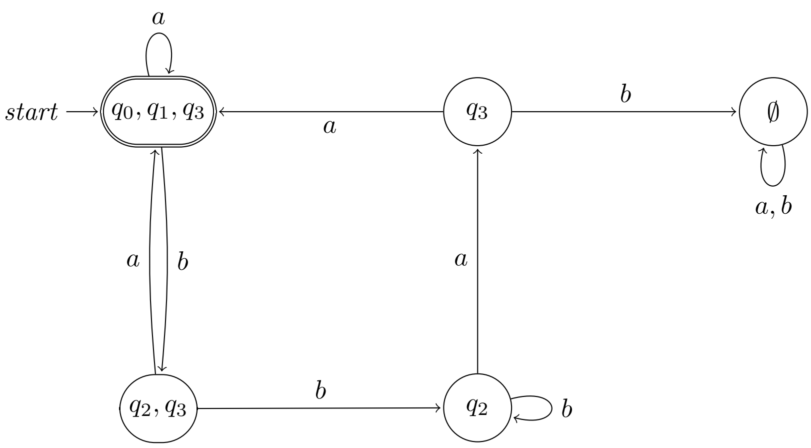
$$
\begin{aligned}
\delta^*(q_0, aabba) &= \delta^*(\{q_0, q_1\}, abba) \\
&= \delta^*(\{q_0, q_1\}, bba) \\
&= \delta^*(\{q_2, q_3\}, ba) \\
&= \delta^*(\{q_2\}, a) \\
&= \delta^*(\{q_3\}, \lambda) \\
&= \{q_3\}
\end{aligned}
$$

So aabba is not accepted.

(b) *aabbaa is accepted (ends in states $\{q_0, q_1, q_3\}$, bbb is not accepted (ends in states $\{q_2\}$.*

(c) λ*-closure*$(q_0) = \{q_0, q_1, q_3\}$.
λ*-closure*$(q_1) = \{q_1, q_3\}$.
λ*-closure*$(q_2) = \{q_2\}$.
λ*-closure*$(q_3) = \{q_3\}$.

(d) *M'' is*



End Answer Info ....

2. In a number of steps, we construct a regular expression e such that

$$\mathcal{L}(e) = L = \{w \in \Sigma^* \mid abba \text{ is not a substring of } w\}.$$

(a) **(5 points)** Give an NFA M such that its language is $\overline{L}$. (NB: This is the *complement* of L.) Explain your answer.

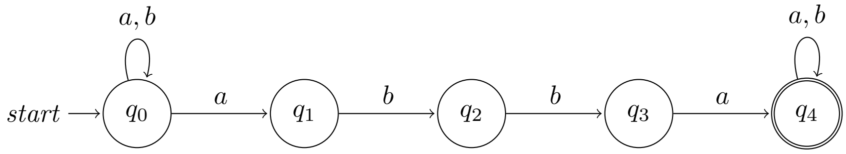
(b) **(5 points)** Construct from M, using the subset construction, a DFA M' accepting the same language. Reduce the number of states in M' by joining states that have the same effect. This yields a DFA M''.

(c) Modify M'' to obtain M''' that accepts L.

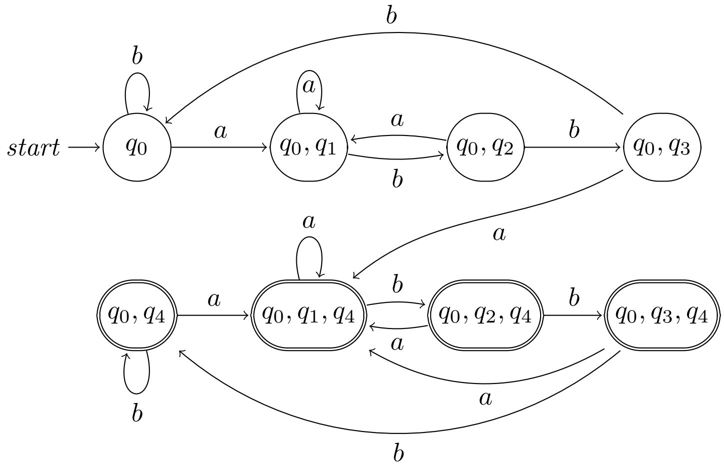
(d) Compute a regular expression e such that $\mathcal{L}(e) = \mathcal{L}(M''')$.

Answer Info: ....

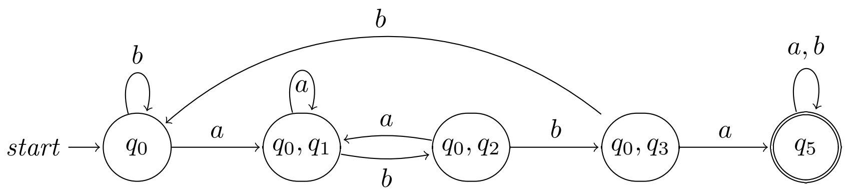
(a) This NFA accepts all words that contain abba as a substring. For these words the automaton can end up in q_4.



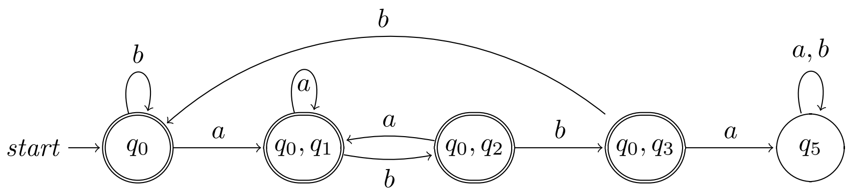
(b) M' is



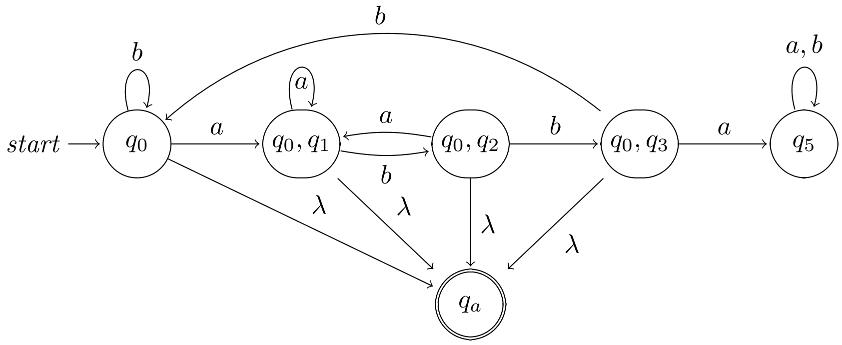
Because from any state that has q_4 all paths lead to accepting states, we can simplify this to M'':



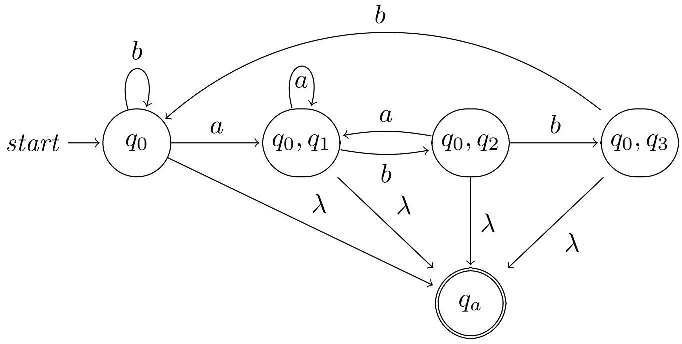
(c) To get a DFA that accepts L, take the complement of accepting states. I.e. all accepting states become non-accepting and vice-versa. This gives M''':



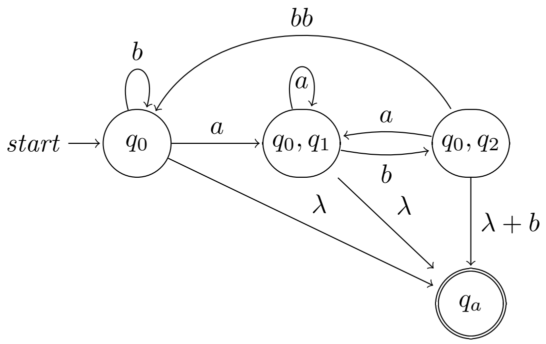
(d) *First add a single accepting state*



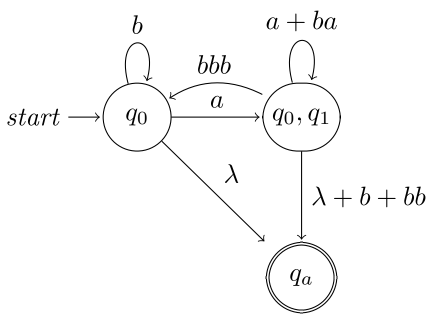
We compute a regular expression by repeatedly deleting states. First delete q_5.



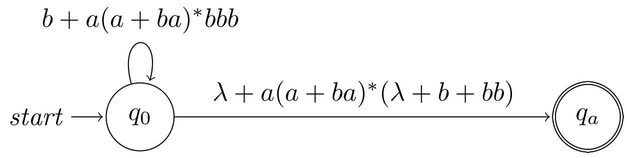
Now delete q_0, q_3*:*



Now delete q_0, q_2*:*



Now delete q_0, q_1*:*



The regular expression corresponding to this automaton is

$$e = (b + a(a + ba)^*bbb)^*(\lambda + a(a + ab)^*(\lambda + b + bb))$$

End Answer Info . . . .

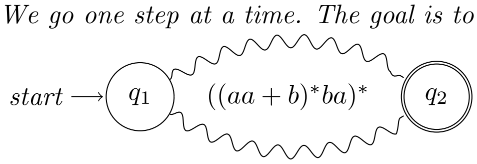
3. Consider the regular expression $e = ((aa + b)^*ba)^*$.

   (a) **(5 points)** Construct an NFA_λ M that accepts the language $\mathcal{L}(e)$, following the steps shown in the lecture (or the steps of the course notes LnA). Explain what you are doing.

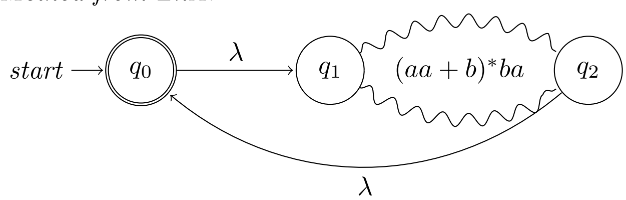
   (b) **(5 points)** Construct a DFA M' out of M that accepts the same language $\mathcal{L}(e)$. Explain what you are doing.

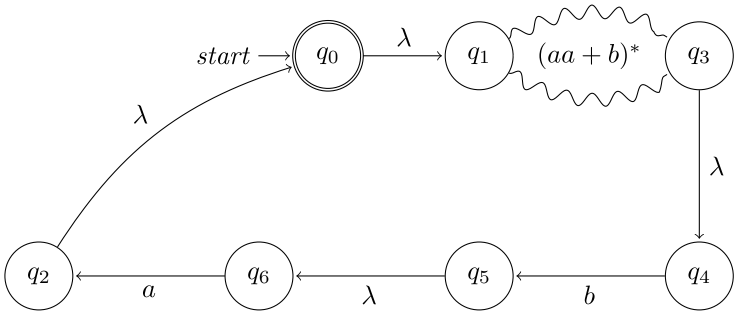
Answer Info: . . . .

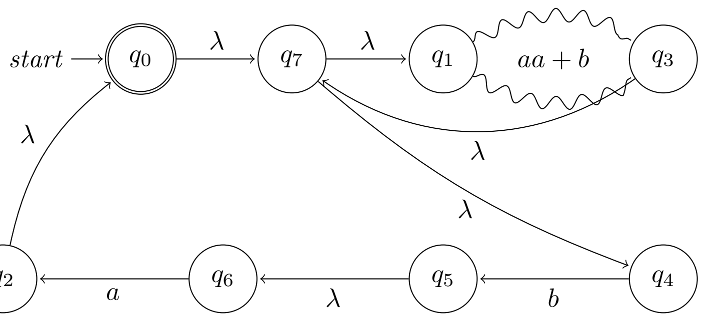
(a) We go one step at a time. The goal is to find the automaton

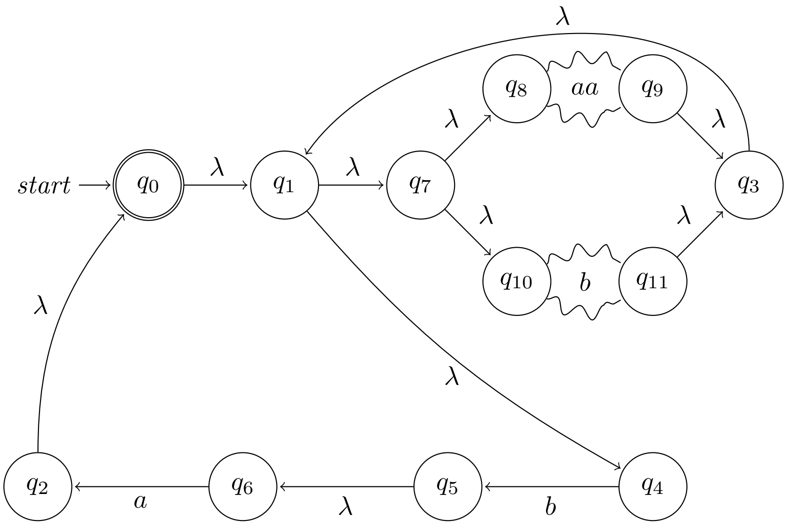


We first do this by following the method from the course notes LnA, and then by following the method from the lecture, which is on the slides.
Method from LnA:

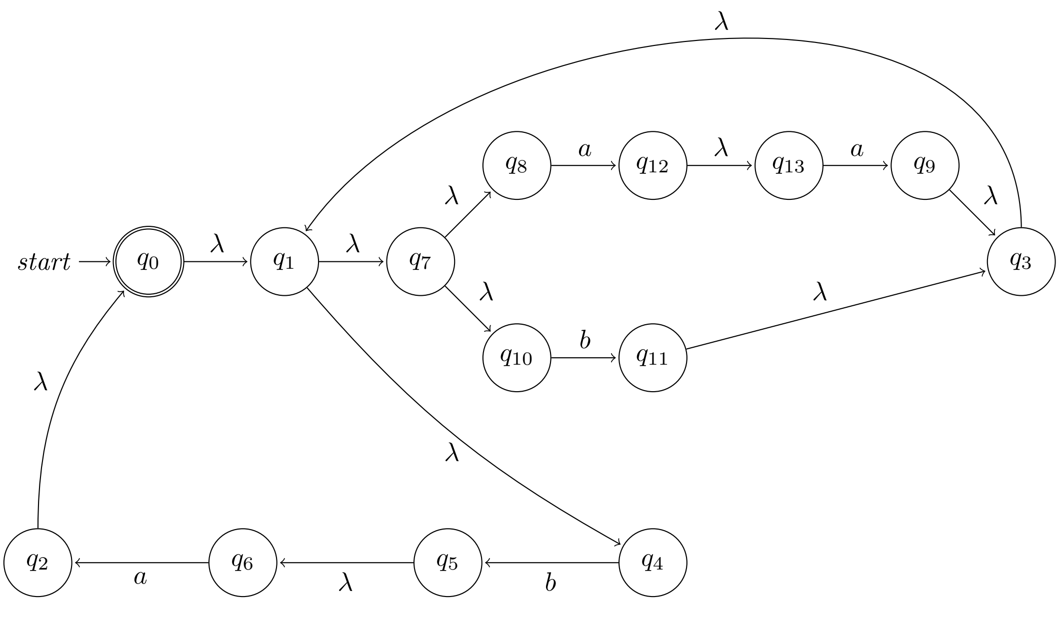




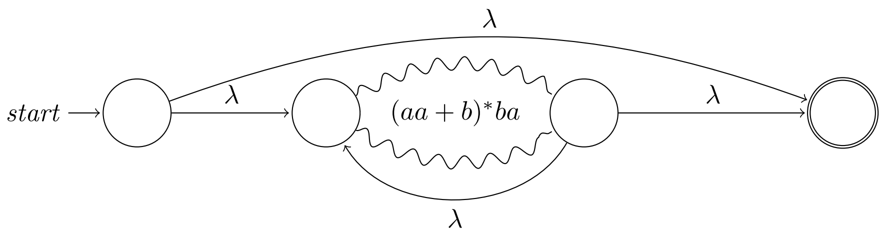




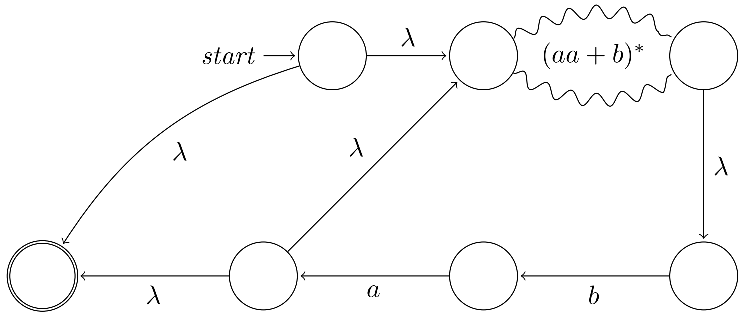
This gives the NFA_λ, M:

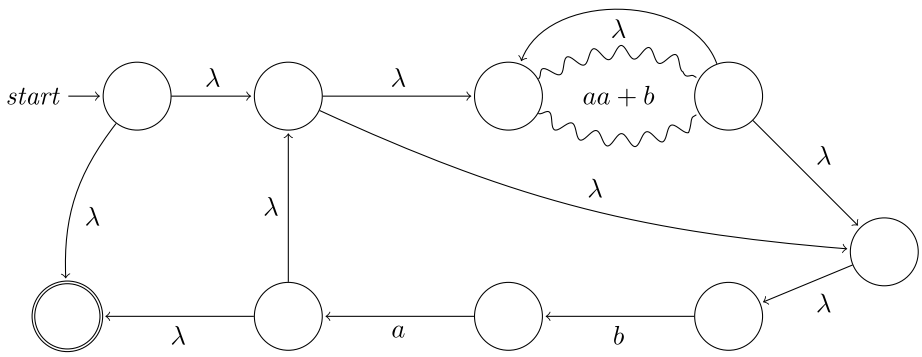


Method from the slides (we don't name the states):

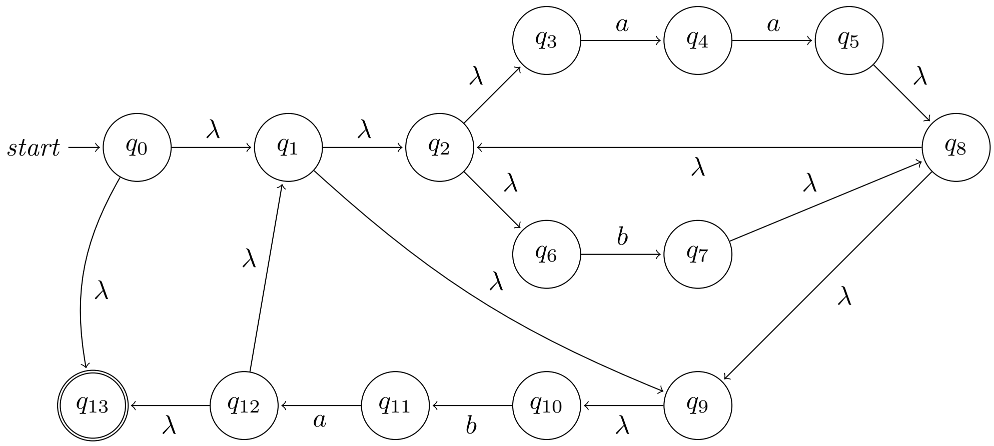


[We omit the λ between the b and a-step.]





This yields the following NFA$_\lambda$, where we now label the states. [We omit the λ between the two a-steps.]



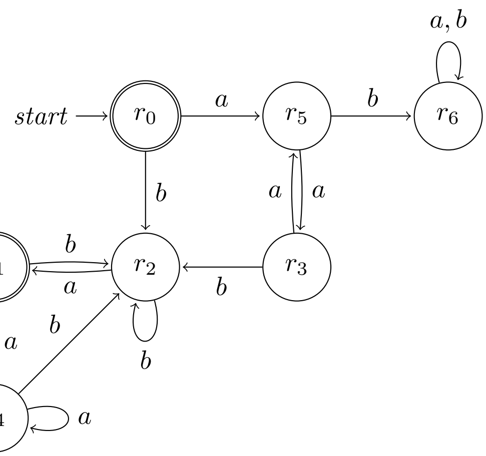
(b) **Following LnS:** *M' is given by the following transition function δ'*

state	a	b
$q_0, q_1, q_4, q_7, q_8, q_{10}$	q_{12}, q_{13}	$q_1, q_3, q_4, q_5, q_6, q_7, q_8, q_{10}, q_{11}$
$q_0, q_1, q_2, q_4, q_7, q_8, q_{10}, q_{12}, q_{13}$	$q_1, q_3, q_4, q_7, q_8, q_9, q_{10}, q_{12}, q_{13}$	$q_1, q_3, q_4, q_5, q_6, q_7, q_8, q_{10}, q_{11}$
$q_1, q_3, q_4, q_5, q_6, q_7, q_8, q_{10}, q_{11}$	$q_0, q_1, q_2, q_4, q_7, q_8, q_{10}, q_{12}, q_{13}$	$q_1, q_3, q_4, q_5, q_6, q_7, q_8, q_{10}, q_{11}$
$q_1, q_3, q_4, q_7, q_8, q_9, q_{10}$	q_{12}, q_{13}	$q_1, q_3, q_4, q_5, q_6, q_7, q_8, q_{10}, q_{11}$
$q_1, q_3, q_4, q_7, q_8, q_9, q_{10}, q_{12}, q_{13}$	$q_1, q_3, q_4, q_7, q_8, q_9, q_{10}, q_{12}, q_{13}$	$q_1, q_3, q_4, q_5, q_6, q_7, q_8, q_{10}, q_{11}$
q_{12}, q_{13}	$q_1, q_3, q_4, q_7, q_8, q_9, q_{10}$	$\emptyset$
$\emptyset$	$\emptyset$	$\emptyset$

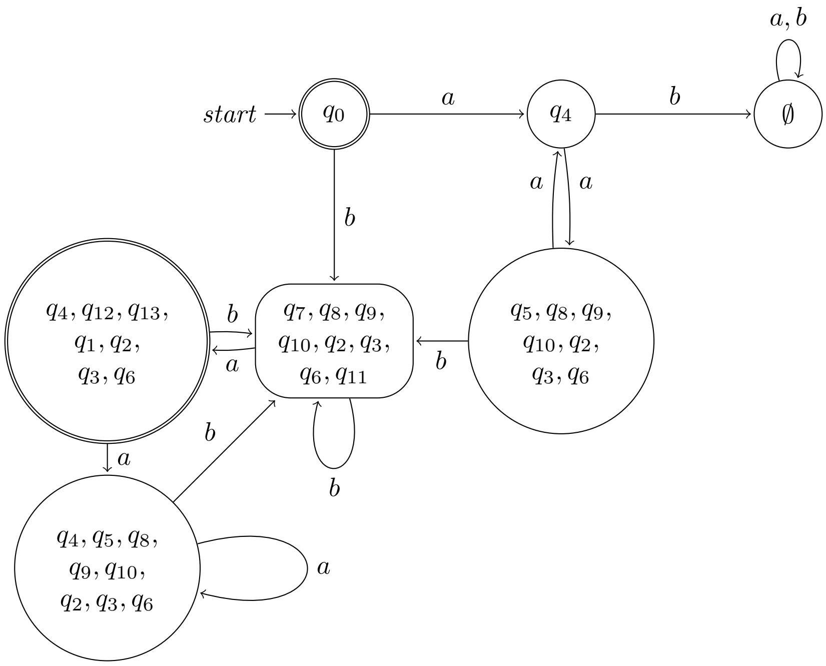
The initial state is $q_0, q_1, q_4, q_7, q_8, q_{10}$*; accepting states are those containing* q_0*. We can relabel this to*

state	a	b
r_0	r_5	r_2
r_1	r_4	r_2
r_2	r_1	r_2
r_3	r_5	r_2
r_4	r_4	r_2
r_5	r_3	r_6
r_6	r_6	r_6

We can draw this DFA as



Following the slides: M' *is as follows*



This is the same as the DFA above.

End Answer Info ....

4. (More challenging) Let L over $\Sigma = \{a, b\}$ be regular. Show that the following language L' is also regular.
$$L' := \{w \in \Sigma^* \mid \exists v \in L(w \text{ is contained in } v)\}$$
NB. w is *contained in* v if all the symbols of w occur in v in the same order, to be precise: $v = s_1 \ldots s_n$ and $w = s_{i_1} \ldots s_{i_m}$ for some sequence $1 \leq i_1 < \ldots < i_m \leq n$. ($s_j \in \Sigma$.)

Answer Info: ....
We know that L is regular, so we can assume (i) that we have a regular expression e for it and (ii) that we have a DFA M for it. Starting from either (i) or (ii) we can show that L' is regular.

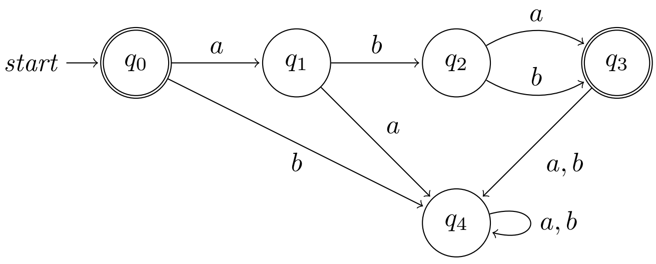
(i) Replace in e every letter a by $(\lambda + a)$, to obtain e'. Then $L' = \mathcal{L}(e')$.

(ii) In M, if there is an arrow from state q to p, add a λ-transition from q to p, obtaining M'. Then $L' = \mathcal{L}(M')$

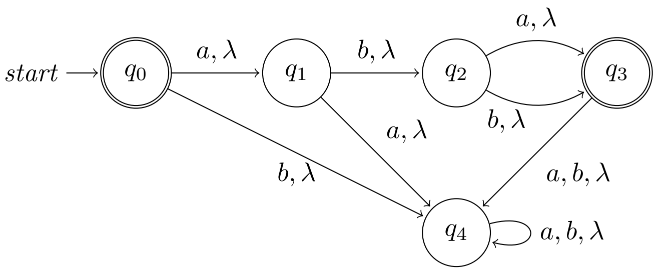
The proofs are omitted here. We do an example

(i) If $L = \mathcal{L}(ab(a + b)ab)$, then $L' = \mathcal{L}((\lambda + a)(\lambda + b)(\lambda + a + b)(\lambda + a)(\lambda + b))$.

(ii) With $L = \mathcal{L}(ab(a + b))$, we have the DFA M:



Now, M' looks like this



End Answer Info . . . .