Homework 3

Ricky Hempel

February 20, 2018

1.4a-c, 1.5b-d, 1.8b, 1.9a, 1.11, 1.14, 1.16, 1.18, 1.21, 1.28

1.4 a.Let the state diagram M_1 recognize $L_1 = \{w \mid \text{w has at least three a's }\}$:

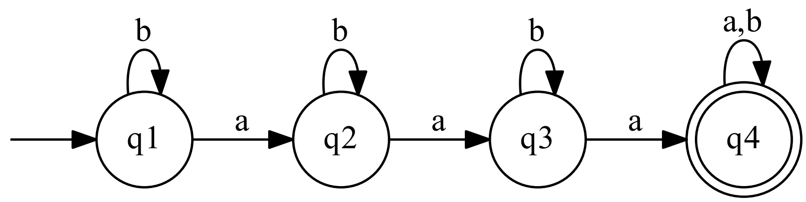


Figure 1: State diagram for w has at least three a's.

Let the state diagram M_2 recognize $L_2 = \{w \mid \text{w has at least twos b's }\}$:

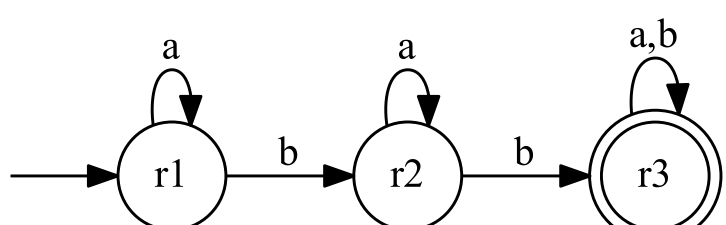


Figure 2: State diagram for w has at least two b's.

The machine M will accept the input if and only if both M_1 and M_2 accept. Because language L is the intersection L_1 and L_2. The state diagram of M that recognizes the language. $L = \{w \mid \text{w has at least three a's and at least two b's}\}$:

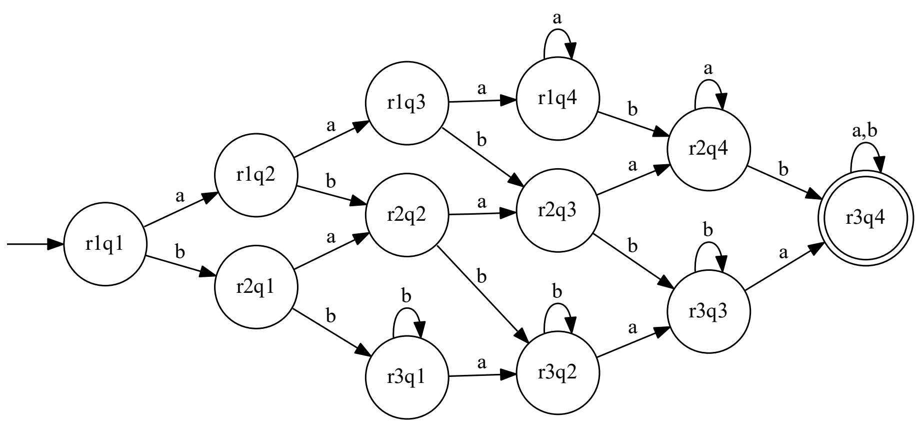


Figure 3: State diagram for w has at least three a's and at least two b's.

b.Let the state diagram M_1 recognize $L_1 = \{w \mid \text{w has exactly two a's }\}$:

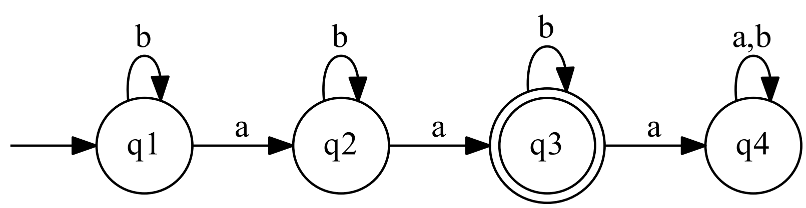


Figure 4: State diagram for w has exactly two a's.

Let the state diagram M_2 recognize $L_2 = \{w \mid \text{w has at least twos b's }\}$:

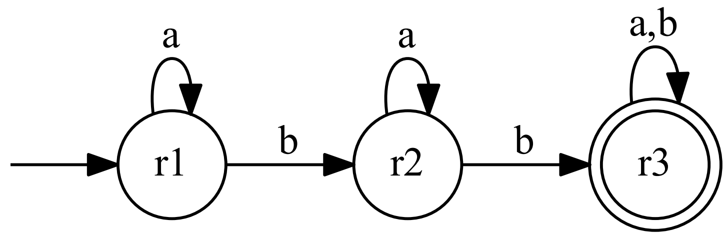


Figure 5: State diagram for w has at least two b's.

The machine M will accept the input if and only if both M_1 and M_2 accept. Because language L is the intersection L_1 and L_2. The state diagram of M that recognizes the language. $L = \{w \mid \text{w has exactly two a's and at least two b's}\}$:

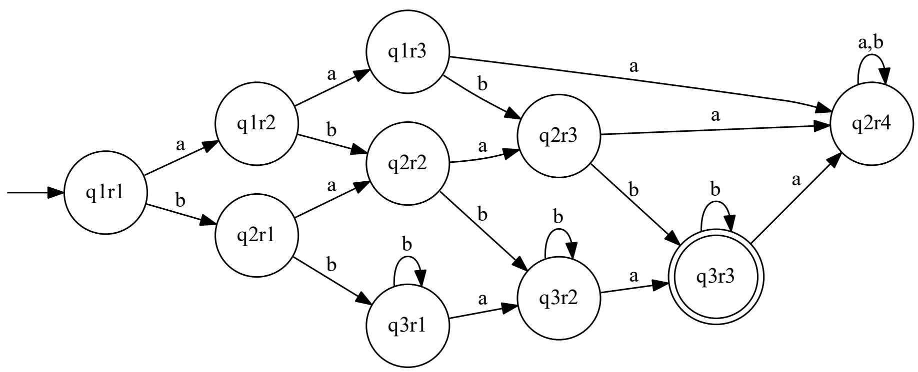


Figure 6: State diagram for w exactly two a's and at least two b's.

c.Let the state diagram M_1 recognize $L_1 = \{w \mid \text{w has an even number of a's }\}$:

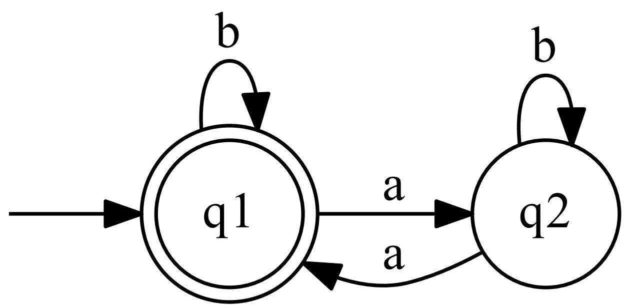


Figure 7: State diagram for w has an even number of a's.

Let the state diagram M_2 recognize $L_2 = \{w \mid \text{w has one or two b's }\}$:

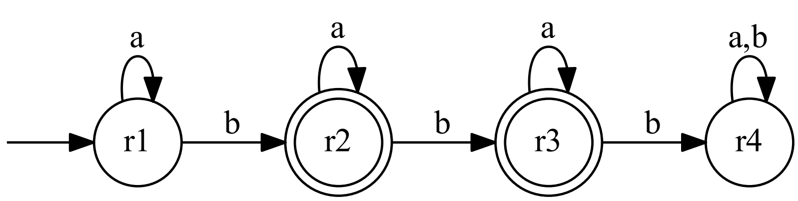


Figure 8: State diagram for w has one or two b's.

The machine M will accept the input if and only if both M_1 and M_2 accept. Because language L is the intersection L_1 and L_2. The state diagram of M that recognizes the language. $L = \{w \mid \text{w has an even number of a's and one or two b's}\}$:

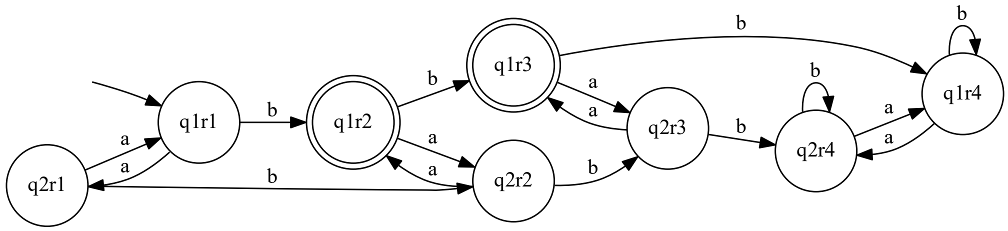


Figure 9: State diagram for w has an even number of a's and at least two b's.

1.5 b. This DFA recognizes $L = \{w \mid \text{w contains the substring baba}\}$:

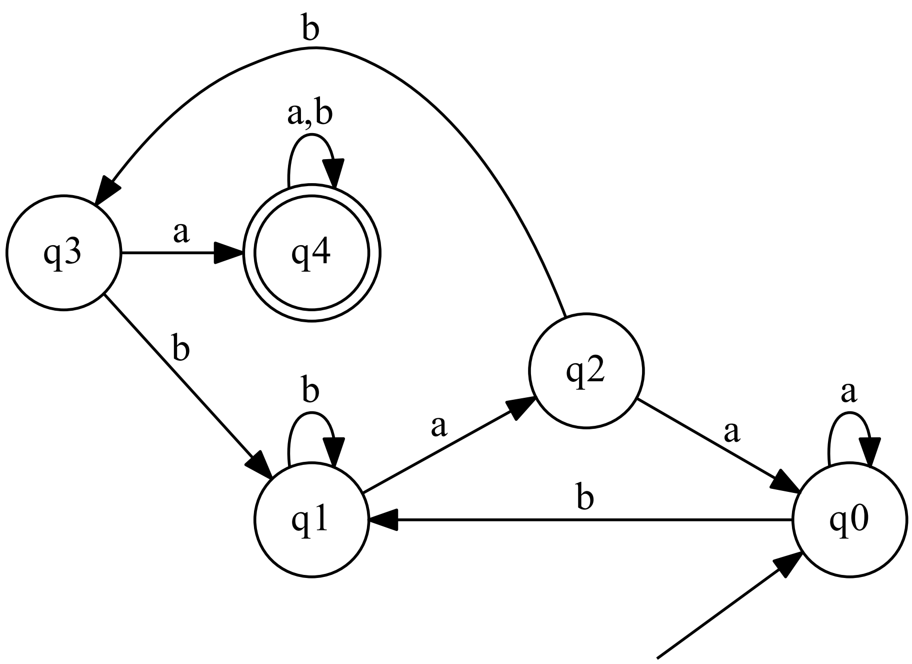


Figure 10: State diagram for w contains the substring baba .

This DFA recognizes $L = \{w \mid \text{w does not contain the substring baba}\}$:

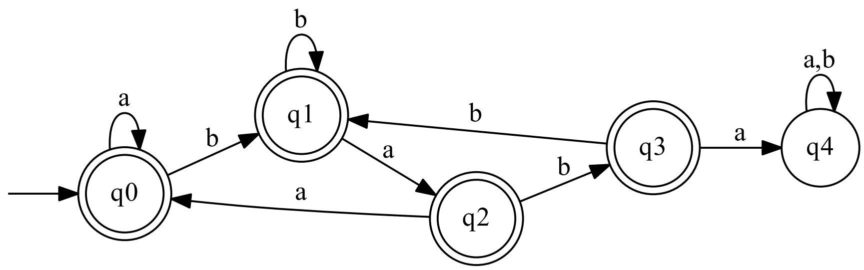


Figure 11: State diagram for w does not contain the substring baba .

c. This DFA recognizes $L = \{w \mid \text{w contains either the substring ab or ba}\}$:

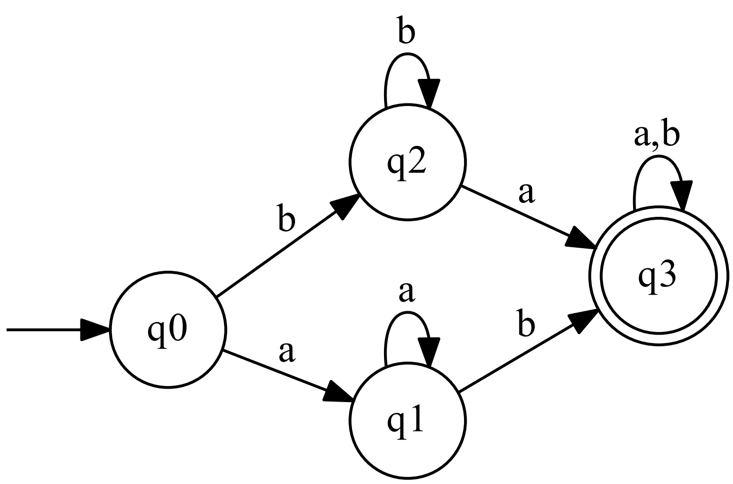


Figure 12: State diagram for w contains either the substring ab or ba .

This DFA recognizes $L = \{w \mid \text{w contains neither the substring ab nor ba }\}$:

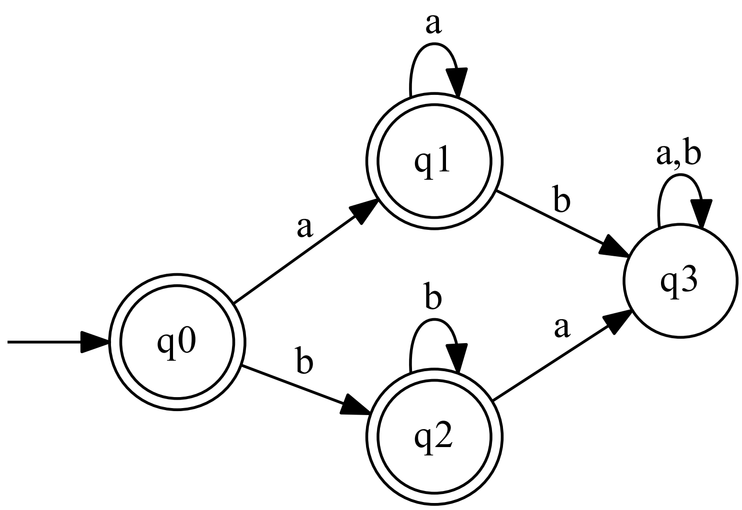


Figure 13: State diagram for w contains neither the substring ab nor ba .

d. This DFA recognizes $L = \{w \mid \text{w is any string not in } a^*b^* \}$:

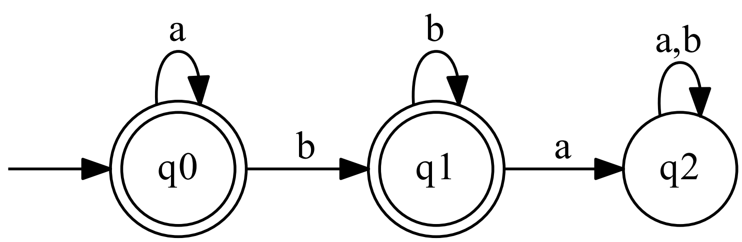


Figure 14: State diagram for w is any string not in a^*b^*

This DFA recognizes $L = \{w \mid \text{w is any string in } a^*b^* \}$:

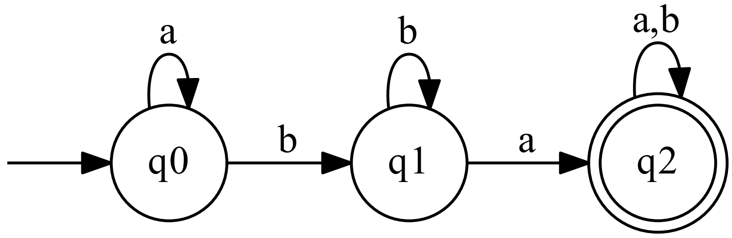


Figure 15: State diagram for w is any string in a^*b^*.

1.8 b.

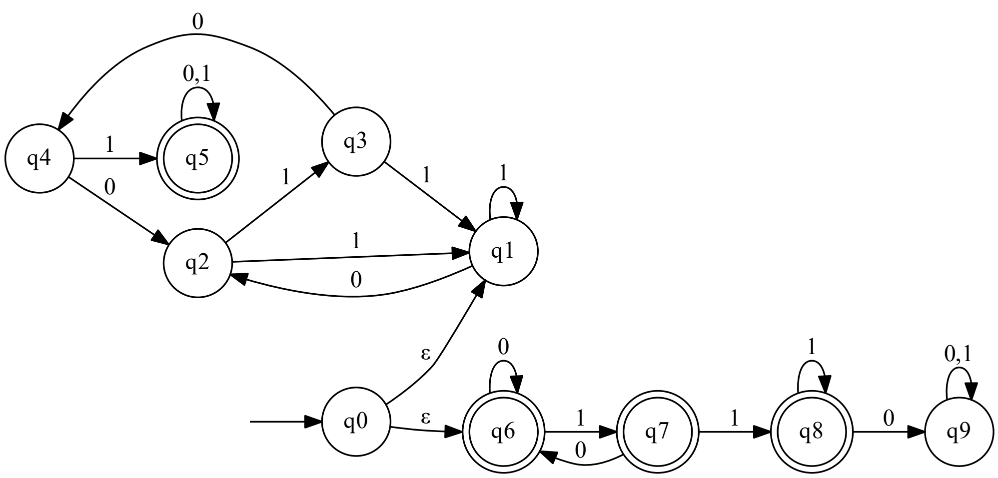


Figure 16: The union of 1.6c and 1.6f.

1.9 a.

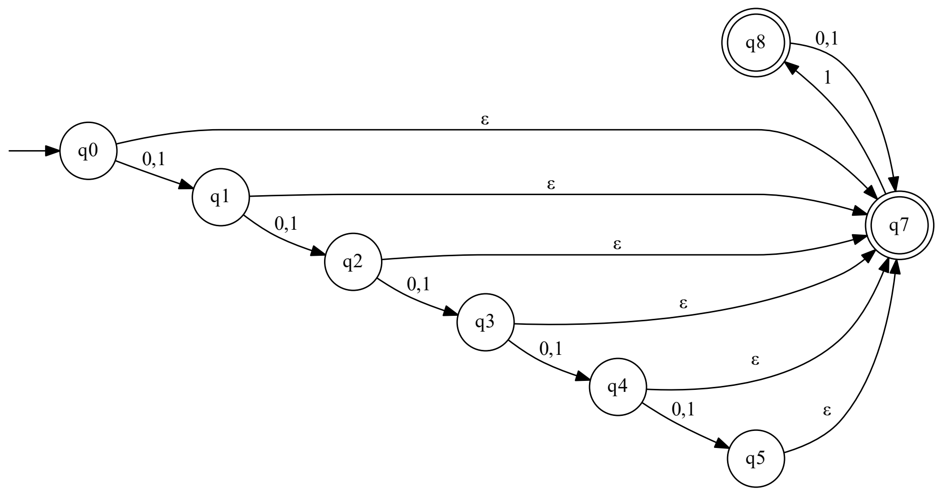


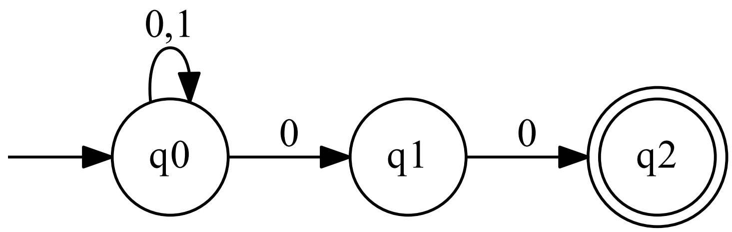
Figure 17: The concatenation of 1.6g and 1.6i.

1.11 *Proof.* Let N= $(Q, \Sigma, \delta, q_0, F)$ be any NFA. Create an NFA $N^{'}$ with a single accept state that recognizes the same language as N. $N^{'}$ is exaclty like N except it has ε-transitions from the states corresponding to the accept states of N, to a new accept state q_{accept}. State q_{accept} has no emerging transitions.$N^{'} = (Q \cup \{q_{accept}\}), \Sigma, \delta^{'}, q_0, \{q_{accept}\})$, where for each $q \in Q$ and $a \in \Sigma$.

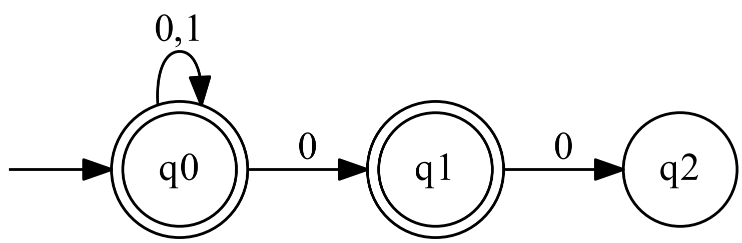
$$\delta^{'} = \begin{cases} \delta(q,a) if a \neq \varepsilon or q \notin F \\ \delta(q,a) \cup \{q_{accept} if a = \varepsilon and q \in F\} \end{cases}$$

and $\delta^{'}(q_{accept}, a) = \emptyset$ for each $a \in \Sigma$ ■

1.14 a.Suppose language B over the alphabet Σ has a DFA M= $(Q, \Sigma, \delta, q_1, F)$.Then the a DFA for the complementary language B^c is $M^c = (Q, \Sigma, \delta, q_1, Q-F)$. The reason why M recognizes M^c is as follows. First note that M and M^c have the same transition function δ. Thus, since M is deterministic,M^c is also deterministic. Now consider any string $w \in \Sigma^*$.Running M on input string w will result in M ending in some state $r \in Q$. Since M is deterministic there is only one possible state that M can end in on input w. If we run M^c on the same input w, then M^c will end in the same state r since M and M^c have the same transition function. Also since M^c is deterministic there is only one possible ending that M^c can be in on input w.
Now suppose that $w \in B$. Then M will accept w, which means that the ending state $r \in F$,i.e, r in an accept state of M. But then r$\notin Q - F$, so M^c does not accept w since M^c has $Q - F$ as its set of accept states. Similarly suppose that $w \notin B$. Then M will not accept w, which means that the ending state $r \notin F$. But then $r \in Q - F$, so M^c accepts w. Therefore M^c accepts string w if and only if M does not accept string w, so M^c recognizes language M^c. Hence the class of regular languages is closed under complement.
b.The NFA M below recognizes the language C= $\{w \in \Sigma^* \mid$ w ends with 00 $\}$, where $\Sigma = \{0, 1\}$.



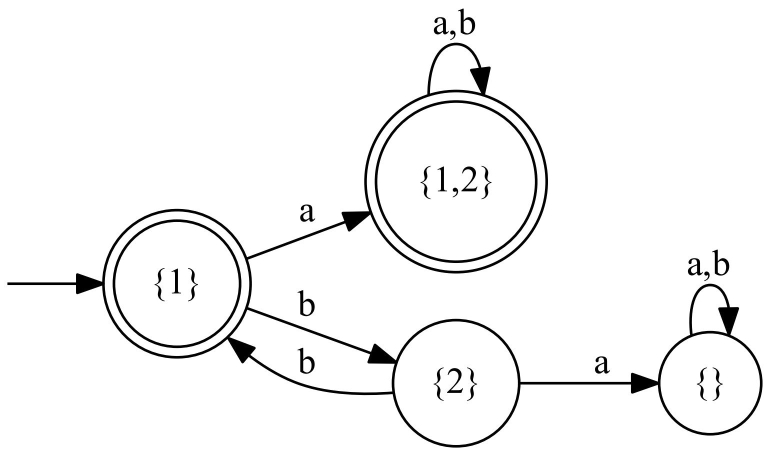
Swapping the accept and non-accept states of M makes the following NFA $M^{'}$.



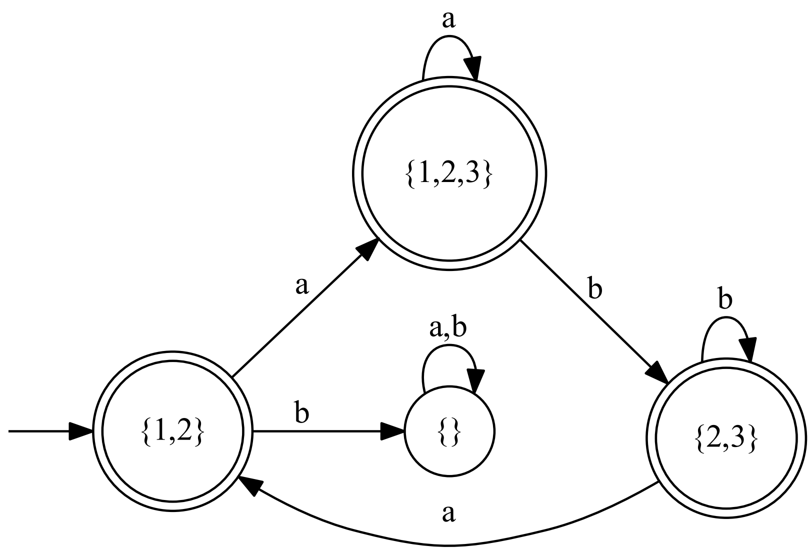
Note that $M^{'}$ accepts the string,100$\notin \bar{C} = \{w \mid \text{does not end with} 00\}$ so $M^{'}$ does not recognize the language $\bar{C}$.
The class of languages recognized by NFAs is closed under complement, which we can as follows. Suppose that C is a recognized by NFA M, i.e., C= $L(M)$. Since every NFA has an equivalent DFA(Theorem 1.19), there is a DFA D such that $L(D) = L(M) = C$. Since every DFA is also an NFA, this then shows that there is an NFA, in particular $\bar{D}$, that recognizes the language $\bar{C} = \bar{L} = (\bar{D})$. Thus, the class of languages recognized by the NFA is closed under complement.

1.16 a.NFA:N= $(Q, \Sigma, \delta, q_0, F)$
DFA:M= $(Q^{'}, \Sigma^{'}, \delta, q_0^{'}, F^{'})$
1.$Q^{'} = P(Q)$
$Q^{'} = \{\emptyset, \{1\}, \{2\}, \{1,2\}\}$
$\Sigma^{'} = \Sigma$
$\delta^{'}(\emptyset, a) = \emptyset$
$\delta^{'}(\emptyset, b) = \emptyset$
$\delta^{'}(\{1\}, a) = \delta(1, a) = \{1, 2\}$
$\delta^{'}(\{1\}, b) = \delta(1, b) = \{2\}$
$\delta^{'}(\{2\}, a) = \delta(2, a) = \emptyset$

$\delta^{'}(\{2\}, b) = \delta(2, b) = \{1\}$
$\delta^{'}(\{1, 2\}, a) = \delta(1, a) \cup \delta(2, a) = \{1, 2\} \cup \emptyset = \{1, 2\}$
$\delta^{'}(\{1, 2\}, b) = \delta(1, b) \cup \delta(2, b) = \{2\} \cup \{1\} = \{1, 2\}$
$q^{'} = \{q_0\}$
$q_0 = \{1\}$
$F^{'} = \{R \in Q^{'} \mid R\text{contains an accept state of N }\}$
$F^{'} = \{\{1\}, \{1, 2\}\}$



b.NFA:N= $(Q, \Sigma, \delta, q_0, F)$
DFA:M= $(Q^{'}, \Sigma^{'}, \delta, q_0^{'}, F^{'})$
1.$Q^{'} = P(Q)$
$Q^{'} = \{\emptyset, \{1\}, \{2\}, \{3\}\{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$
$\Sigma^{'} = \Sigma$
$\delta^{'}(\emptyset, a) = \emptyset$
$\delta^{'}(\emptyset, b) = \emptyset$
$\delta^{'}(\{1, 2\}, a) = \delta(\{1\}, a) \cup \delta(\{2\}, a) = \{1, 2, 3\}$
$\delta^{'}(\{1, 2\}, b) = \delta(\{1\}, a) \cup \delta(\{2\}, b) = \emptyset$
$\delta^{'}(\{2, 3\}, a) = \delta(\{2\}, a) \cup \delta(\{3\}, a) = \{1, 2\}$
$\delta^{'}(\{2, 3\}, b) = \delta(\{2\}, b) \cup \delta(\{3\}, b) = \{2, 3\}$
$\delta^{'}(\{1, 2, 3\}, a) = \delta(\{1\}, a) \cup \delta(\{2\}, a) \cup (\{3\}, a) = \{1, 2, 3\}$
$\delta^{'}(\{1, 2, 3\}, b) = \delta(\{1\}, b) \cup \delta(\{2\}, b) \cup (\{3\}, b) = \{2, 3\}$
$q^{'} = \{q_0\}$
$q_0 = \{1, 2\}$
$F^{'} = \{R \in Q^{'} \mid R\text{contains an accept state of N }\}$
$F^{'} = \{\{1, 2\}, \{2, 3\}, \{1, 2, 3\}\}$



1.18 Let $\Sigma = (0 \cup 1)$ and $\Sigma^* = (0 \cup 1)^*$

a.$1\Sigma^*0$

b.$\Sigma^*1\Sigma^*1\Sigma^*1\Sigma^*$

c.$\Sigma^*0101\Sigma^*$

d.$\Sigma\Sigma0\Sigma^*$

e.$(0 \cup 1\Sigma)(\Sigma\Sigma)^*$

f.$0^*(10^+)^*1^*$

g.$(\varepsilon \cup \Sigma)^5$

h.$\varepsilon \cup \Sigma \cup 0\Sigma \cup 10 \cup 0\Sigma\Sigma \cup 10\Sigma \cup 110 \cup \Sigma^3\Sigma^+$

i.$(1\Sigma)^*(\varepsilon \cup 1)$

j.$00^+ \cup 100^+ \cup 0^+10^+ \cup 00^+1$

k.$0 \cup \varepsilon$

l.$1^*(01^*01^*) \cup 0^*10^*10^*$

m.$\emptyset$

n.Σ^+

1.21 a.$a^*b(a \cup ba^*b)^*$

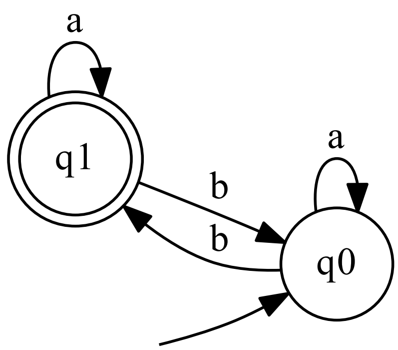


Figure 18: The starting finite automata.

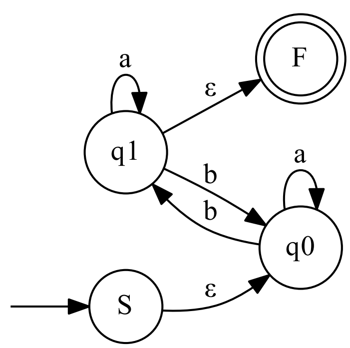


Figure 19: The adding a new start and accept states.

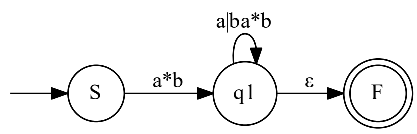


Figure 20: After removing q0.

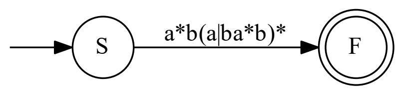


Figure 21: After removing q1.

b.$\varepsilon \cup (a \cup b)a^*b((a(a \cup b) \cup b)a^*b)^*(\varepsilon \cup a)$

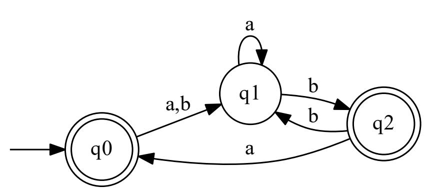


Figure 22: The starting finite automata.

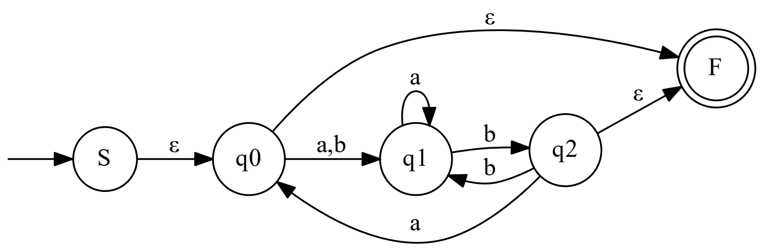


Figure 23: After adding a new start and end state.

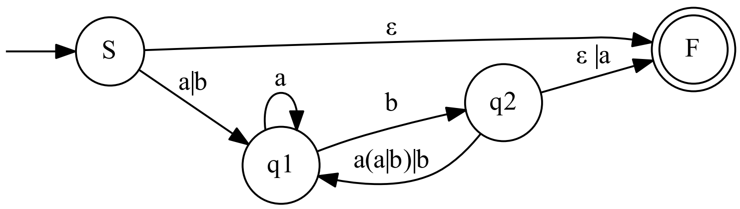


Figure 24: After removing q0.

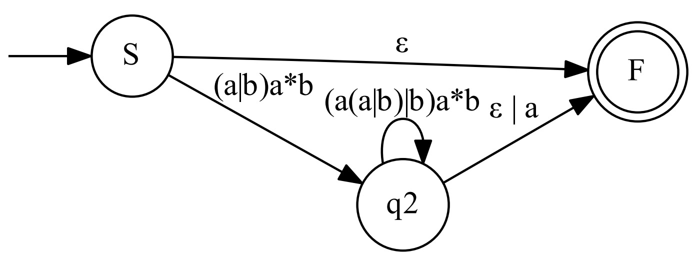


Figure 25: After removing q1.

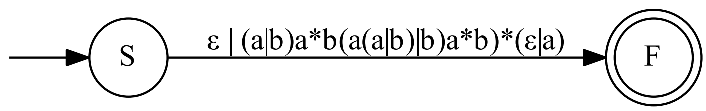


Figure 26: After removing q2.

1.28 a.

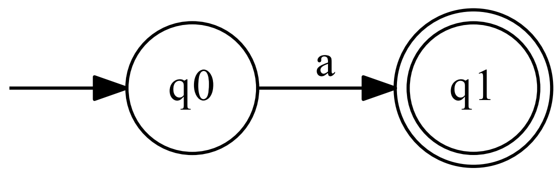


Figure 27: The NFA for a.

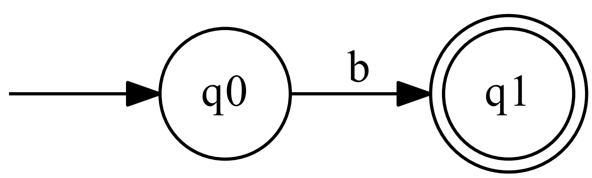


Figure 28: The NFA for b.

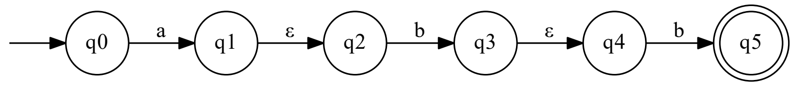


Figure 29: The NFA for abb.

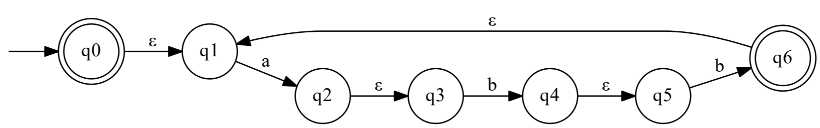


Figure 30: The NFA for (abb)*.

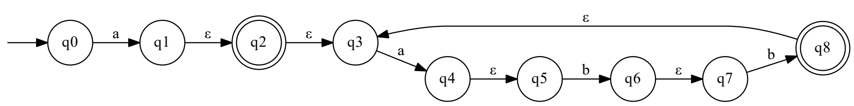


Figure 31: The NFA for a(abb)*.

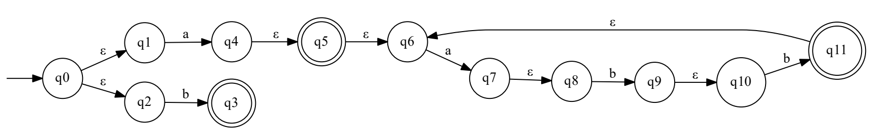


Figure 32: The NFA for a(abb*)∪b.

b.

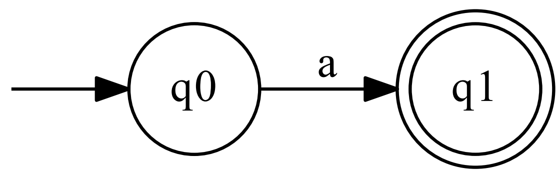


Figure 33: The NFA for a.

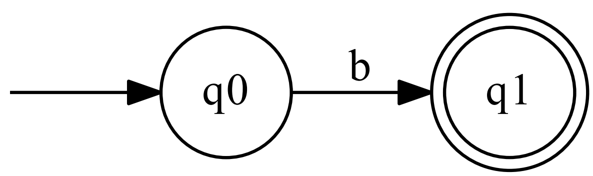


Figure 34: The NFA for b.

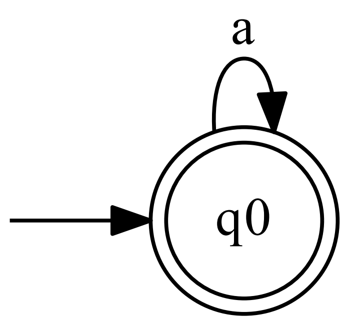


Figure 35: The NFA for a*.

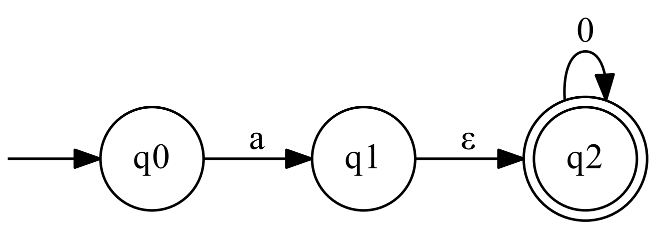


Figure 36: The NFA for a^+.

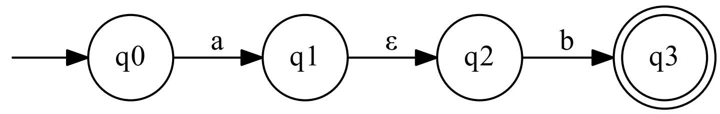


Figure 37: The NFA for ab.

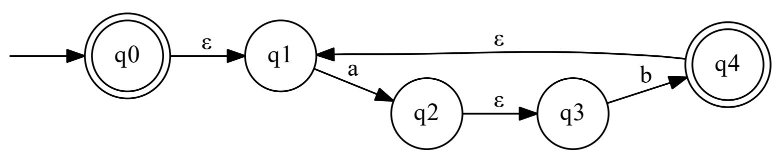


Figure 38: The NFA for (ab)*.

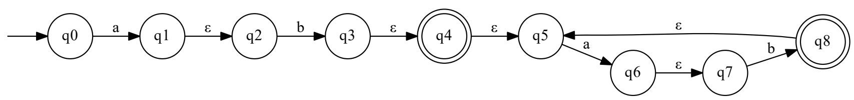


Figure 39: The NFA for $(ab)^+$.

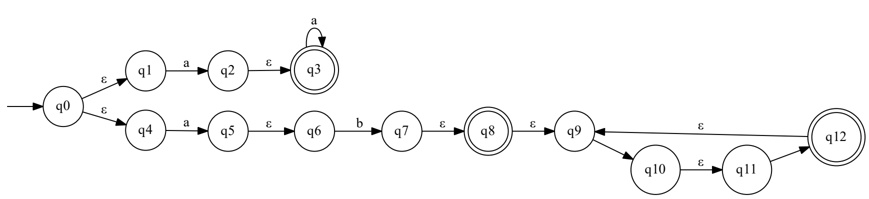


Figure 40: The NFA for $a^+ \cup (ab)^+$.

c.

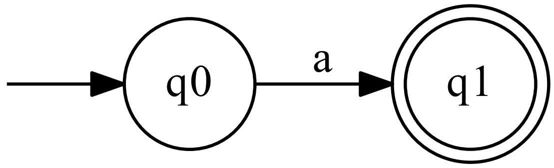


Figure 41: The NFA for a.

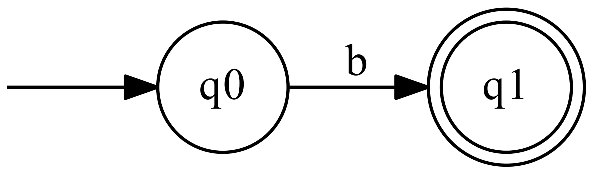


Figure 42: The NFA for b.

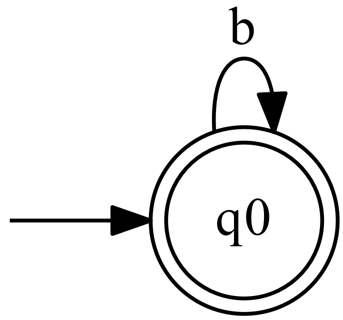


Figure 43: The NFA for b*.

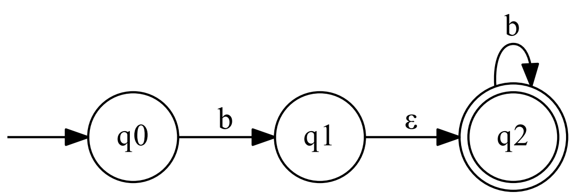


Figure 44: The NFA for b^+.

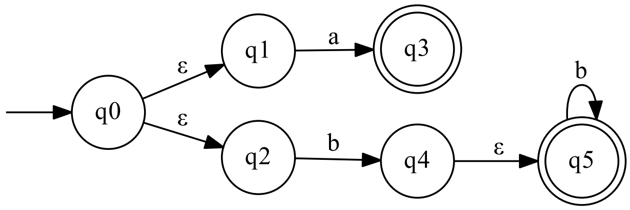


Figure 45: The NFA for $a \cup b^{+}$.

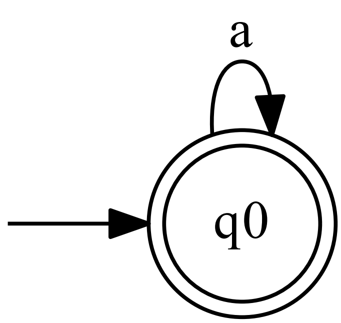


Figure 46: The NFA for a*.

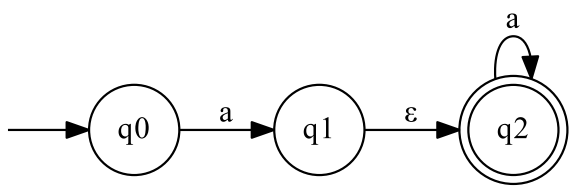


Figure 47: The NFA for a^+.

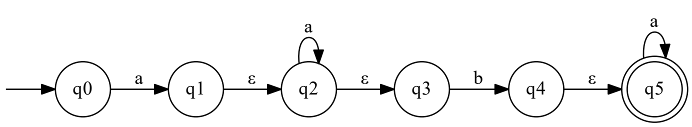


Figure 48: The NFA for a^+b^+.

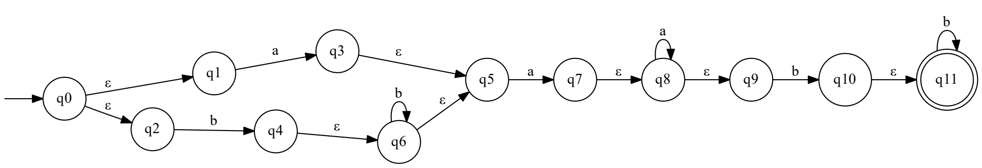


Figure 49: The NFA for $(a \cup b^+)a^+b^+$.