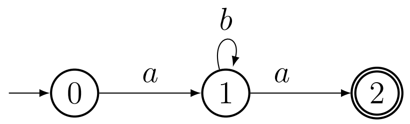
Aufgabe 1



(a) $\Sigma = \{a, b\}$, $Q = \{0, 1, 2\}$, $q_0 = 0$, $F = \{2\}$

$\delta(0, a) = 1 \quad \delta(1, a) = 2 \quad \delta(2, a) = -$

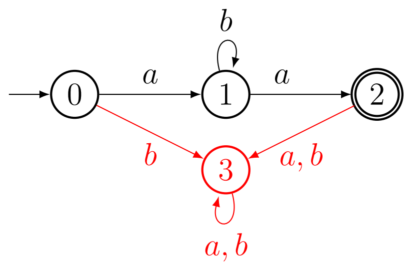
$\delta(0, b) = - \quad \delta(1, b) = 1 \quad \delta(2, b) = -$

Automaten sollten nur in Ausnahmefällen implizit definierte Zustandswechsel wie hier enthalten. Siehe dazu die Bemerkung in der Lösung von (d).

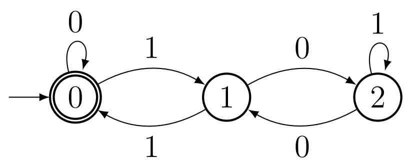
(b) ε (false), ab (false), $abbbba$ (true), bab (false)

(c) Das Zeichen a, gefolgt von 0, 1, 2, … Zeichen b, gefolgtvon einem Zeichen a.

(d) Ein solcher „unvollständiger Automat“ sollte nur dann verwendet werden, wenn er durch viele Übergänge in nicht akzeptierende Zustände unübersichtlich wird.



Aufgabe 2



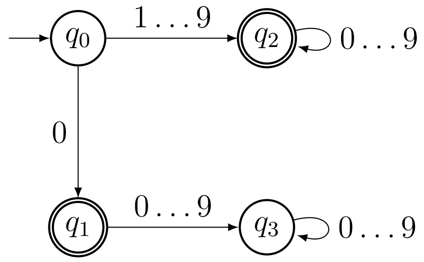
dec	bin	akzeptiert
0	0	ja
1	1	nein
2	10	nein
3	11	ja
4	100	nein
5	101	nein
6	110	ja
7	111	nein

dec	bin	akzeptiert
8	1000	nein
9	1001	ja
10	1010	nein
11	1011	nein
12	1100	ja
13	1101	nein
14	1110	nein
15	1111	ja

Der Automat akzeptiert genau die Binärzahlen, welche durch 3 teilbar sind.

Aufgabe 3

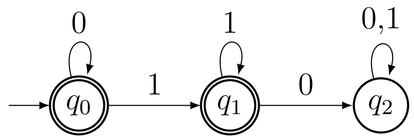
DFA über $\Sigma = \{0, 1, \ldots, 9\}$, der alle nichtnegativen ganzen Zahlen ohne führende Nullen akzeptiert:



Beachte, dass eine *einzelne* Null in einem separaten Zustand (q_1) akzeptiert wird aber alle Wörter, die aus mehr als einem Zeichen bestehen und mit einer Null beginnen, gleich nach dieser ersten Null in einem nicht akzeptierenden Zustand (q_3) landen müssen, aus dem es „kein Entkommen“ gibt.

Aufgabe 4

Ein DFA, der alle Wörter akzeptiert, die aus einer (möglicherweise leeren) Folge von Nullen, gefolgt von einer (möglicherweise leeren) Folge von Einsen bestehen:



Auch hier muss man mit dem Zustand q_2 (einer „Senke“) dafür sorgen, dass ein Wort nicht akzeptiert wird, selbst wenn der Automat die ersten Zeichen akzptiert, aber danach noch Zeichen folgen, welche die Bedingung an die Sprache des Automaten verletzen.