

Aufgabenübersicht

1) Endliche Automaten	2
2) Kellerautomat	3
3) Schaltwerk	4
4) XWizard	5

Aufgabe 1

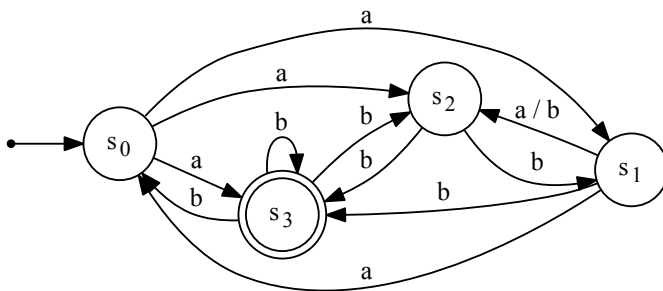
2018-B-01

Endliche Automaten

Gegeben sei der folgende **nichtdeterministische** endliche Automat:

$$A = (\{a, b\}, \{s_0, \dots, s_3\}, \delta, s_0, \{s_3\})$$

δ :



	<i>a</i>	<i>b</i>
<i>s</i> ₀	{ <i>s</i> ₁ , <i>s</i> ₂ , <i>s</i> ₃ }	∅
<i>s</i> ₁	{ <i>s</i> ₀ , <i>s</i> ₂ }	{ <i>s</i> ₂ , <i>s</i> ₃ }
<i>s</i> ₂	∅	{ <i>s</i> ₁ , <i>s</i> ₃ }
<i>s</i> ₃	∅	{ <i>s</i> ₀ , <i>s</i> ₂ , <i>s</i> ₃ }

Erstellen Sie mithilfe des aus der Vorlesung bekannten Algorithmus einen **deterministischen** endlichen Automaten $A' = (E', S', \delta', s'_0, F')$ mit $L(A') = L(A)$ und geben Sie diesen vollständig an.

Hinweis: Geben Sie insbesondere ein Zustandsüberförungsdiagramm an. Nutzen Sie die vorgegebene Tabelle.

	<i>a</i>	<i>b</i>
<i>s</i> ₀		
<i>s</i> ₁		
<i>s</i> ₂		
<i>s</i> ₃		

$$A' = \left(\underbrace{\quad}_{E'}, \underbrace{\quad}_{S'}, \delta', \underbrace{\quad}_{s'_0}, \underbrace{\quad}_{F'} \right)$$

δ' :

Aufgabe 2

2018-B-02

Kellerautomat

Gegeben sei die Sprache aller Wörter w mit

$$L = \{w \in \{a, b, c\}^* \mid w = a^m b^n c^k, \text{ mit } m, n \in \mathbb{N}, k \in \{m, n\}\}.$$

Das heißt, dass in einem Wort der Sprache L zuerst m -mal a , dann n -mal b geschrieben wird; auf diese Zeichenfolge folgt dann entweder m -mal oder n -mal c .

Es gilt beispielsweise:

$$\begin{aligned} aaabc, aaabccc, abbcc, aabc, &\in L, \\ \lambda, aaa, bb, ac, abb, ababc, aabbc, abbbcc &\notin L. \end{aligned}$$

(a) Entwerfen Sie einen **nichtdeterministischen Kellerautomaten**

$$A = \left(\{a, b, c\}, \underbrace{\boxed{s_0, s_1, s_2, s_3, s_e}}_S, \underbrace{\boxed{}}_K, \delta, \underbrace{\boxed{s_0}}_{s_0}, k_0, \underbrace{\boxed{s_e}}_F \right)$$

welcher die Sprache L erkennt. Der erste Teil des Kellerautomaten ist Ihnen im Folgenden bereits vorgegeben und erkennt alle Worte der Form $w = a^m b^n c^m$. Ergänzen Sie diesen Automaten, so dass er zusätzlich auch die Worte der Form $a^m b^n c^n$ erkennt.

$$\begin{aligned} (s_0, a, k_0) &\Rightarrow (s_1, ak_0) \\ (s_1, a, a) &\Rightarrow (s_1, aa) \\ (s_1, b, a) &\Rightarrow (s_2, a) \\ (s_2, b, a) &\Rightarrow (s_2, a) \\ (s_2, c, a) &\Rightarrow (s_3, \lambda) \\ (s_3, c, a) &\Rightarrow (s_3, \lambda) \\ (s_3, \lambda, k_0) &\Rightarrow (s_e, k_0) \end{aligned}$$

(b) Ergänzen Sie die zusätzlich benötigten Übergänge unter der Annahme, dass $m, n \in \mathbb{N}_0$.

Es gilt also beispielsweise zusätzlich

$$\lambda, aacc, bbb, bbcc \in L.$$

Aufgabe 3

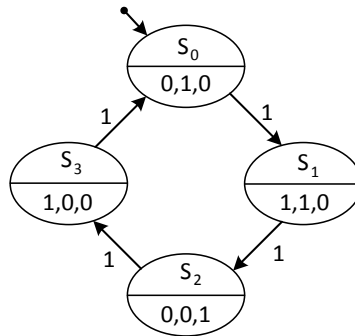
2018-B-03

Schaltwerk

Der gegebene Moore-Automat A beschreibt eine Ampelschaltung, bei der die Lichter der Ampel direkt durch die Signale q_{rot} , q_{gelb} und q_{gruen} an-, bzw. ausgeschaltet werden ($0 \Rightarrow$ aus, $1 \Rightarrow$ an).

$$A = \left(\underbrace{\{1\}}_{\cong \text{Takt}}, \underbrace{\{0, 1\}^3}_{\cong (q_{gelb}, q_{rot}, q_{gruen})}, \{s_0, \dots, s_3\}, \delta, \gamma, \{s_0\} \right)$$

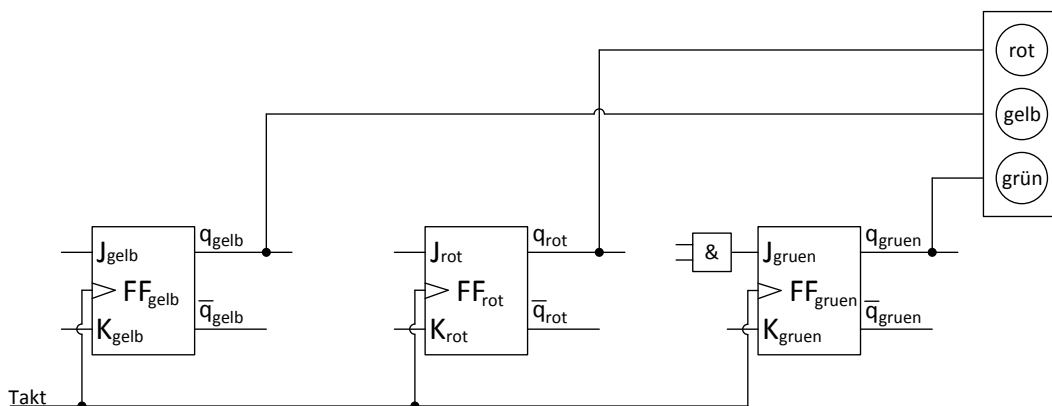
δ, γ :



Verbinden Sie die Eingänge der drei J - K -Flip-Flops in geeigneter Weise mit den Ausgangssignalen q_{rot} , q_{gelb} und q_{gruen} , sodass das Verhalten des Schaltwerks dem Automaten entspricht, also die Lichtfolge rot \rightarrow rot/gelb \rightarrow grün \rightarrow gelb \rightarrow rot \rightarrow u.s.w. entsteht. Füllen Sie dafür auch die gegebene Tabelle aus. Sie benötigen keine zusätzlichen Gatter.

Hinweis: Es kann vorkommen, dass die Belegung eines Eingangssignals für ein Flip-Flop sowohl 0 als auch 1 sein darf, z.B. wenn eine Eins am Ausgang sowohl durch Setzen dieser Eins ($J = 1, K = 0$), als auch durch Speichern von $q^* = 1$ aus dem vorhergehenden Takt ($J = K = 0$) erreicht werden kann. Tragen Sie in diesem Fall ein X in die Tabelle ein.

q_{gelb}	q_{rot}	q_{gruen}	q_{gelb}^*	q_{rot}^*	q_{gruen}^*	J_{gelb}	K_{gelb}	J_{rot}	K_{rot}	J_{gruen}	K_{gruen}
0	1	0	1	0	0						
1	1	0	0	1	0						
0	0	1	1	1	0						
1	0	0	0	0	1						



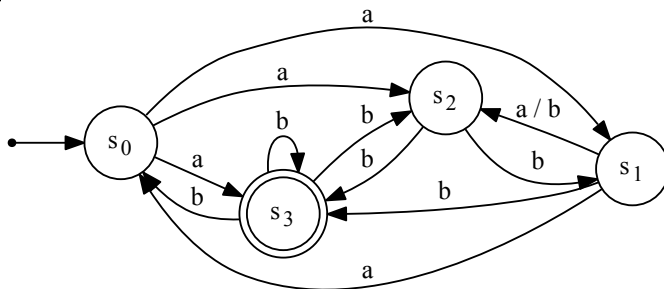
Aufgabe 4

2018-B-04

XWizard

$$A = (\{a, b\}, \{s_0, \dots, s_3\}, \delta, s_0, \{s_3\})$$

δ :



```
fsm:
(s0, a) => s3 | s1 | s2;
(s1, a) => s2 | s0;
(s1, b) => s3 | s2;
(s2, b) => s3 | s1;
(s3, b) => s0 | s2 | s3;
--declarations--
simulateToStep=-1;
input=null;
s0=s0;
F=s3;
--declarations-end--
```

Zum oben links abgebildeten Automaten A aus Aufgabe 1 gehöre das oben rechts abgebildete XWizard-Skript $A_{XS\text{kript}}$.

- (a) Beschreiben Sie kurz, wie Sie vorgehen würden, um A mit XWizard deterministisch und minimal zu machen.

- (b) Beschreiben Sie kurz, wie Sie vorgehen würden, um A mit XWizard auf dem Wort *abaabaa* für drei Schritte zu simulieren.

- (c) Woran sieht man im Skript, dass A nicht deterministisch ist? Welche kennzeichnende Eigenschaft hätte ein Skript eines deterministischen Automaten?

Hinweise:

- Sie dürfen bei (a) und (b) auf Skript- **oder** GUI-Ebene beschreiben.
- Bei (c) müssen Sie das deterministische Skript **nicht** angeben, nur seine wesentliche Eigenschaft beschreiben.
- Die Aufgabe ist einfach, denken Sie nicht zu kompliziert!