

Aufgabenübersicht

0) Statistik (ohne Wertung)	1
1) Minimierung endlicher Automaten	2
2) Turingmaschinen	3
3) Schaltwerke	4

Aufgabe 0

2017-B-00

Statistik (ohne Wertung)

Bitte beantworten Sie diese Frage zum Lehrbuch *Theoretische Informatik* – ganz praktisch:

Wahr Falsch

Ich habe (**auch**) **das Lehrbuch** zur Vorbereitung auf die Bonusklausur genutzt.

Vielen Dank!

Aufgabe 1

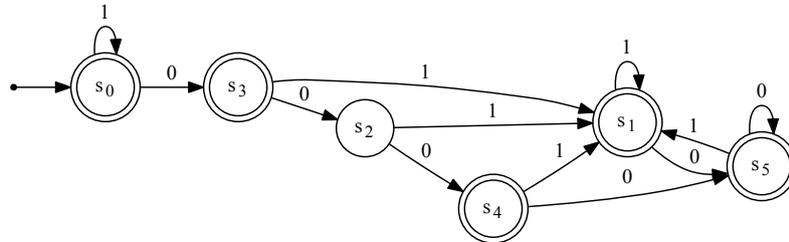
2017-B-01

Minimierung endlicher Automaten

Gegeben sei der deterministische endliche Automat:

$$A = (\{0, 1\}, \{s_0, \dots, s_5\}, \delta, s_0, \{s_0, s_1, s_3, s_4, s_5\})$$

δ :



Füllen Sie die folgende Tabelle nach dem Minimierungsalgorithmus aus der Vorlesung aus.

Lösung:

s_1	\times_2				
s_2	\times_0	\times_0			
s_3	\times_1	\times_1	\times_0		
s_4	\times_2	–	\times_0	\times_1	
s_5	\times_2	–	\times_0	\times_1	–
	s_0	s_1	s_2	s_3	s_4

SKRIPT ID-18927



Hinweis: Sie müssen den minimierten Automaten **nicht** angeben (er ist Teil der Aufgabenstellung von Aufgabe 3).

Aufgabe 2

2017-B-02

Turingmaschinen

Gegeben sei die deterministische Turingmaschine

$$T = (\{a, b\}, \{a, b, A, B, \star\}, \{s_0, s_1, s_2, s_3, s_e\}, \delta, s_0, \{s_e\})$$

mit

δ	a	b	A	B	\star
s_0	(s_1, A, R)			(s_3, B, R)	(s_e, \star, N)
s_1	(s_1, a, R)	(s_2, B, L)		(s_1, B, R)	
s_2	(s_2, a, L)		(s_0, A, R)	(s_2, B, L)	
s_3				(s_3, B, R)	(s_e, \star, N)
s_e					

Erklären Sie **kurz** in eigenen Worten die grundlegende Funktionsweise von T . Geben Sie insbesondere die Sprache $L(T)$ an (formal oder umgangssprachlich präzise).

Hinweis: Es könnte helfen, zunächst ein Wort der Länge vier zu finden, das von T erkannt wird.

Lösung:

$$L(T) = \{a^n b^n \mid n \in \mathbb{N}_0\}$$

Erklärung: Die Turingmaschine überprüft für jedes ‚a‘, ob ein entsprechendes ‚b‘ existiert. Hierbei werden jene a und b, die bereits verarbeitet wurden, mit ‚A‘ und ‚B‘ markiert und gleichzeitig überprüft, dass kein b vor einem a vorkommt. Zustand s_0 markiert das nächste a. Mit Zustand s_1 wird das zugehörige b gesucht und markiert. In s_2 traversiert die Turingmaschine zurück zum nächsten a, das noch nicht verarbeitet wurde. In Zustand s_3 wird überprüft, ob die komplette Bandinschrift abgearbeitet wurde.

SKRIPT ID-19039



Aufgabe 3

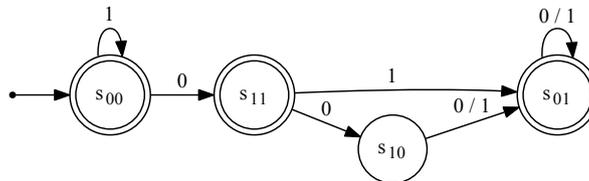
2017-B-03

Schaltwerke

Gegeben sei eine minimierte Version des endlichen Automaten aus Aufgabe 1:

$$A' = (\{0, 1\}, \{s_{00}, \dots, s_{11}\}, \delta', s_0, \{s_{00}, s_{01}, s_{11}\})$$

δ' :



SKRIPT ID-18941



Das umseitig angegebene Schaltwerk soll an Ausgang a genau dann 1 anliegen haben, wenn sich A' für die über e eingegebene Eingabefolge $w \in \{0, 1\}^*$ in einem Endzustand befunden hätte. Es soll also gelten:

$$a = 1 \iff w \in L(A') = L(A)$$

Hinweise:

- Das Schaltwerk ist schon fast fertig. Sie müssen nur entscheiden, ob die insgesamt 17 Eingänge der Gatter **negiert** (zeichnen Sie einen Kringel \circ) oder **unnegiert** (ziehen Sie den Strich durch) angesteuert werden sollen.
- Für Zustand s_{xy} des Automaten soll der Wert x durch Flipflop RS_0 und der Wert y durch RS_1 gespeichert werden.
- Sie können in die Tabellen (die **nicht** gewertet werden) die Ausgangswerte q_0, q_1 der Flipflops RS_0, RS_1 in Abhängigkeit der Eingabe e und der Ausgangswerte aus dem vorherigen Schritt q_0^*, q_1^* sowie den Ausgangswert der Schaltung a eintragen.
- Um den Punkt für diese Aufgabe zu erhalten, müssen **alle Verbindungen** in der Schaltung richtig gesetzt werden.

e	q_0^*	q_1^*	q_0	q_1
0	0	0	1	1
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	0
1	0	1	0	1
1	1	0	0	1
1	1	1	0	1

q_0	q_1	a
0	0	1
0	1	1
1	0	0
1	1	1

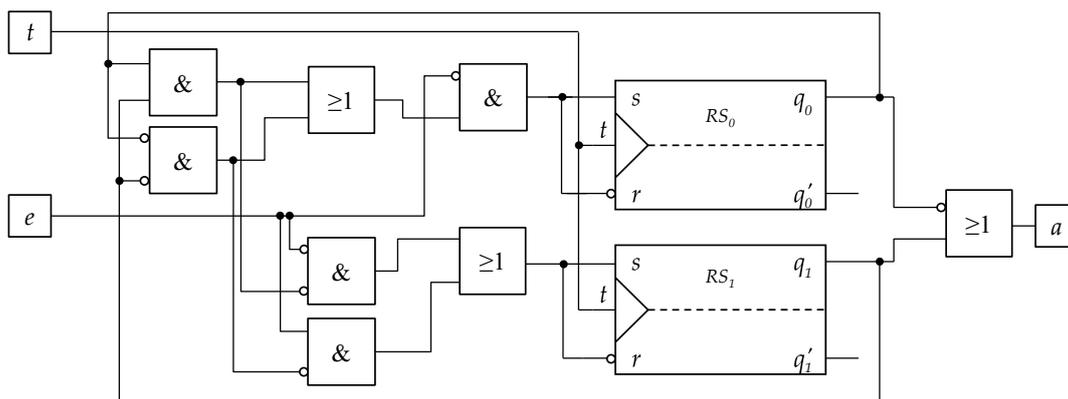
Lösung: Zunächst füllen wir die Tabellen entsprechend der Übergänge von A' aus. Von s_{00} (was $q_0^* = 0$ und $q_1^* = 0$ entspricht) landen wir bei Eingabe $e = 0$ im nächsten Schritt bei s_{11} (was $q_0 = 1$ und $q_1 = 1$ entspricht); also tragen wir in der ersten Tabelle in der ersten Zeile 1|1 ein. So gehen wir für alle Felder vor. In der zweiten Tabelle tragen wir genau dann 1 ein, wenn bei der entsprechenden Kombination (q_0, q_1) der Zustand $s_{q_0q_1}$ ein Endzustand ist, also immer, außer bei $q_0 = 1$ und $q_1 = 0$. Nun lesen wir aus der ersten Tabelle die Booleschen Terme ab:

$$q_0 = \neg e \neg q_0^* \neg q_1^* \vee \neg e q_0^* q_1^* \quad \text{Nach Einsen in der Tabelle abgelesen}$$

$$q_1 = (e \vee \neg q_0^* \vee \neg q_1^*) \wedge (\neg e \vee q_0^* \vee q_1^*) \quad \text{Nach Nullen in der Tabelle abgelesen}$$

$$= e(q_0^* \vee q_1^*) \vee \neg e(\neg q_0^* \vee \neg q_1^*) \quad \text{Ausgeklammert oder scharfer Blick}$$

Etwa die obere Hälfte der Schaltung berechnet den Wert q_0 des oberen Flipflops: e muss auf jeden Fall negiert sein, also haben wir ganz rechts einen Kringel in dem oberen Eingang des UND-Gatters. Die Flipflopwerte aus dem vorherigen Schritt müssen dabei entweder beide $q_0^* = q_1^* = 0$ oder beide $q_0^* = q_1^* = 1$ sein, was den Rest der oberen Hälfte festlegt. Die untere Hälfte berechnet den Wert q_1 des unteren Flipflops. Scharfen Blickes sieht man (auch schon direkt in der Tabelle, vgl. die beiden Nullen in der Mitte), dass q_1 genau dann 1 ist, wenn e wenigstens gleich q_0^* **oder** q_1^* ist. Also müssen wir nur diese beiden Fälle umsetzen und am Ende mit einem ODER verknüpfen. Kleiner Trick: die UND-Gatter aus der oberen Hälfte müssen als NAND genutzt werden (Kringel am nächsten Eingang), man muss also die beiden ODER in der Klammer oben jew. in ein NAND umsetzen durch $x \vee y \iff \neg x \text{ NAND } \neg y$.



Den „kleinen Trick“ sollten Sie kennen – er ist auch bei CMOS-Aufgaben sehr wichtig. Es handelt sich dabei um eine der beiden grundlegenden DeMorgan-Regeln.

Aus der zweiten Tabelle lesen wir schließlich ab, dass a als $\neg q_0 \vee q_1$ berechnet werden kann – ein Kringel und zwei durchgezogene Linien beim Schaltungsteil rechts, und wir sind fertig.