Prof. Dr. Jasmin Blanchette
Jannis Limperg
Luca Maio

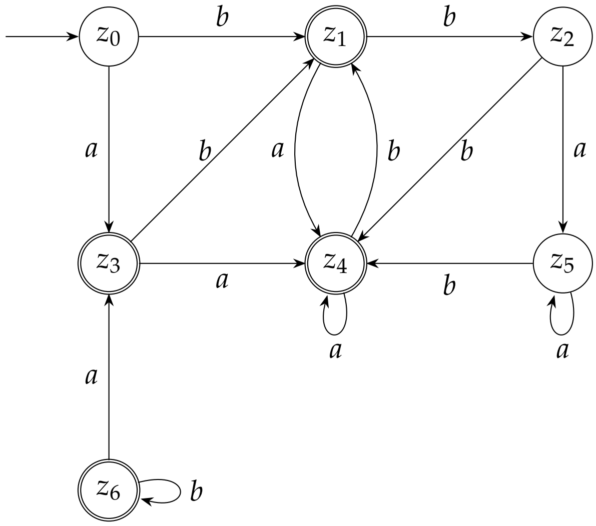
Ludwig-Maximilians-Universität München
Institut für Informatik
Abgabe bis 26. Juni 2024, 10:00 Uhr

Lösungsvorschlag zur Übung 9 zur Vorlesung
Theoretische Informatik für Studierende der Medieninformatik

Wenn Sie Automaten angeben, tun Sie dies immer in Form eines Zustandsgraphen. Andere Formen der Darstellung (z.B. als Liste von Übergängen) werden nicht gewertet, da sie sehr viel aufwändiger zu korrigieren sind. Vergessen Sie nicht, im Zustandsgraph Start- und Endzustände zu markieren.

TIMI9-1 ***Algorithmen Wiederholung*** (2 Punkte)

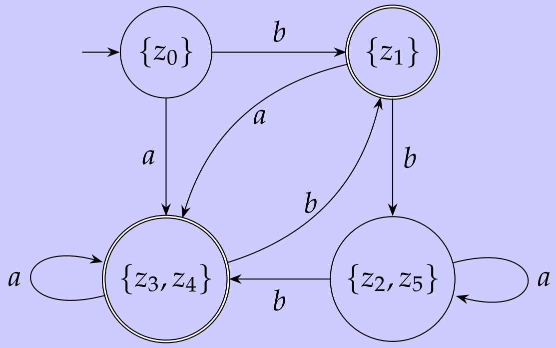
a) Minimieren Sie den folgenden DFA. Verwenden Sie die tabellarische Variante des Algorithmus zur Minimierung von DFAs aus der Vorlesung (nicht die graphische Variante und nicht den Algorithmus von letztem Jahr). Geben Sie die Partitionstabelle und den minimalen DFA an.



LÖSUNGSVORSCHLAG:
Wir entfernen den unerreichbaren Zustand z_6.
Partitionstabelle:

$$
\begin{array}{l}
z_0 \quad z_2 \quad z_5 \;\Big|\; z_1 \quad z_3 \quad z_4 \\
z_0 \;\Big|\; z_2 \quad z_5 \;\Big|\; z_1 \quad z_3 \quad z_4 \quad \text{mit } a \\
z_0 \;\Big|\; z_2 \quad z_5 \;\Big|\; z_1 \;\Big|\; z_3 \quad z_4 \quad \text{mit } b
\end{array}
$$

Aus der Partitionierung ergibt sich der Minimalautomat:



Sei G die Grammatik $(\{A_1, A_2, A_3, A_4, A_5\}, \{\$, \#\}, P, A_1)$ mit

$$\begin{aligned} P = \{ & A_1 \to A_2A_5 \mid A_3A_5, \\ & A_2 \to A_4A_3 \mid \$, \\ & A_3 \to A_2A_5, \\ & A_4 \to \$, \\ & A_5 \to A_5A_1 \mid \#\} \end{aligned}$$

Prüfen Sie mit dem CYK-Algorithmus, ob das Wort \$##\$## in $L(G)$ ist. Erstellen Sie dazu die entsprechende Tabelle des Algorithmus und erklären Sie anhand der Tabelle, ob das Wort in $L(G)$ ist.

LÖSUNGSVORSCHLAG:

Wort:	\$	#	#	\$	#	#
j\ i	1	2	3	4	5	6
1	A_2, A_4	A_5	A_5	A_2, A_4	A_5	A_5
2	A_1, A_3			A_1, A_3		
3	A_1		A_5	A_1		
4			A_5			
5	A_1					
6	A_1					

Da das Startsymbol A_1 in Zeile 6, Spalte 1 enthalten ist, ist \$##\$## $\in L(G)$.

TIMI9-2 ***Kontextfreie Sprachen*** (0 Punkte)

Sei L die formale Sprache aller Wörter $w \in \{a, b\}^*$, sodass w in der zweiten Hälfte mindestens ein b enthält:

$$L = \{ubv \mid u, v \in \{a, b\}^*, |u| > |v|\}$$

a) Geben Sie eine kontextfreie Grammatik an, die L erzeugt.

LÖSUNGSVORSCHLAG:
$G = (\{S, U, V\}, \{a, b\}, P, S)$ mit

$$P = \{S \to USV \mid Ub, U \to a \mid b \mid UU, V \to a \mid b\}$$

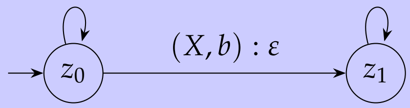
Erläuterung: S erzeugt eine Satzform der Form $U^i b V^{i-1}$. Dann können beliebig viele weitere U links von b erzeugt werden mit $U \to UU$, wobei die Anzahl an V gleich bleibt, damit gilt $|u| > |v|$. Die Nichtterminale U und V können dann beliebig a's und b's erzeugen.

b) Geben Sei einen Kellerautomaten an, der L akzeptiert. Entsprechend der Definition aus der Vorlesung soll Ihr Kellerautomat durch leeren Keller akzeptieren.

Hinweis: Nutzen Sie den Nichtdeterminismus des Automaten aus.

LÖSUNGSVORSCHLAG:

$(\#, a) : X\#$	$(\#, \varepsilon) : \varepsilon$
$(\#, b) : X\#$	$(X, \varepsilon) : \varepsilon$
$(X, a) : XX$	$(X, a) : \varepsilon$
$(X, b) : XX$	$(X, b) : \varepsilon$



Erläuterung: Im Startzustand z_0 werden beliebige a und b gelesen, wobei jeweils ein X pro Zeichen in den Keller gelegt wird. Der Nichtdeterminismus wird verwendet, um das richtige b zu lesen und in den Zustand z_1 zu wechseln. Mit diesem Wechseln können nur noch soviele a und b gelesen werden, wie zuvor gelesen wurden, da jedes Lesen ein X vom Keller abbaut. Folgen weniger Zeichen, so werden die verbleibenden X durch ε-Übergänge abgebaut. Das Startsymbol im Keller wird durch einen weiteren ε-Übergang entfernt.