Prof. Javier Esparza
Kevin Kappelmann, Simon Roßkopf, Lukas Stevens

Technische Universität München
Lehrstuhl für Theoretische Informatik

Einführung in die Theoretische Informatik

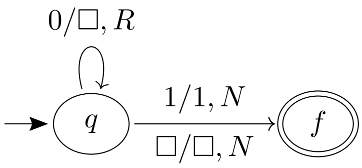
Sommersemester 2023 – Übungsblatt 9

- Das Übungsblatt ist in zwei Teile gegliedert: den Vorbereitungsteil, den Sie vor der Übung selbstständig bearbeiten sollen, und den Übungs-/Nachbereitungsteil, der Aufgaben enthält, die in der Übung besprochen werden und von Ihnen anschließend zur Nachbereitung verwendet werden können.
- Das ist nicht das Hausaufgabenblatt! Die Hausaufgaben finden Sie auf einem separaten Blatt.

Notation von TMs: Bei Turing Maschinen verwendet man eine zu PDAs ähnliche graphische Notation: Sei $M = (\{q, f\}, \{0, 1\}, \{0, 1, \Box\}, \delta, q, \Box, \{f\})$ eine TM mit δ:

$$\begin{aligned} \delta(q, \Box) &= (f, \Box, N) \\ \delta(q, 0) &= (q, \Box, R) \\ \delta(q, 1) &= (f, 1, N) \end{aligned}$$

Diese Maschine entfernt führende Nullen. Nun schreibt man auf die Transitionen $\alpha/\beta, D$ mit $\alpha, \beta \in \Gamma$ und $D \in \{L, N, R\}$. Dies bedeutet, dass der Bandbuchstabe α an der Kopf Position steht und durch β ersetzt wird. Danach bewegt sich der Kopf nach links (L), rechts (R) oder gar nicht (N). Graphisch ist dies:



Vorbereitung (→ vor der Übung selbständig zu bearbeiten)

Individualaufgabe Ü9.1. (*Wichtige Begriffe*)

Überprüfen Sie, dass Sie die folgenden Begriffe oder Notationen korrekt definieren können.

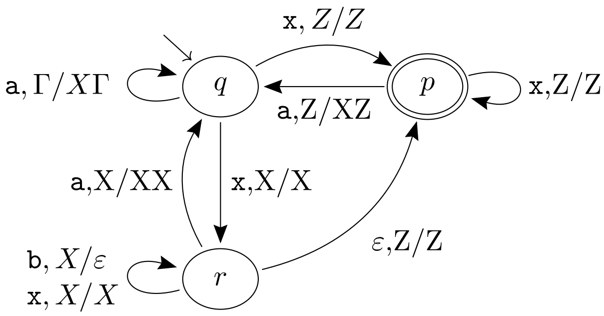
- Konversion CFG ↔ PDA
- deterministischer Kellerautomat (DPDA)
- deterministische kontextfreie Sprache (DCFL)
- Abschlusseigenschaften von DCFL
- Abschlusseigenschaften von CFL
- abzählbar / überabzählbar
- berechenbar / unberechenbar
- Church-Turing These
- nichtdeterministische / deterministische Turing-Maschine (TM)
- Konfiguration einer TM
- akzeptierte Sprache einer TM
- Turing-berechenbar

Individualaufgabe Ü9.2. (*Automata Tutor: "DPDAs"*)

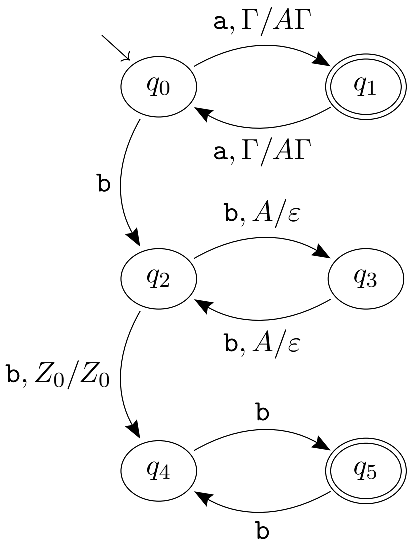
Lösen Sie die Aufgaben Ü9.2 (a–b) auf Automata Tutor.[1] **Achtung:** Ihr konstruierter Automat darf nicht zu viele Zustände haben (siehe Aufgabenstellung). Wenn Sie einen ε-Übergang angeben wollen, geben Sie statt ε bitte `E` ein (siehe Hinweisbox über Canvas) und beim Entfernen eines Kellersymbols, lassen Sie den Teil hinter "/" bitte leer (Beispiel: "$\mathtt{a}, X/$"). Die Simulation ist nicht aktiviert.

Lösungsskizze.

(a) Zu der Aufgabe gibt es ein Lösungsvideo. Mögliche Lösung:



(b) Mögliche Lösung:



Individualaufgabe Ü9.3. (*Meine erste TM*)

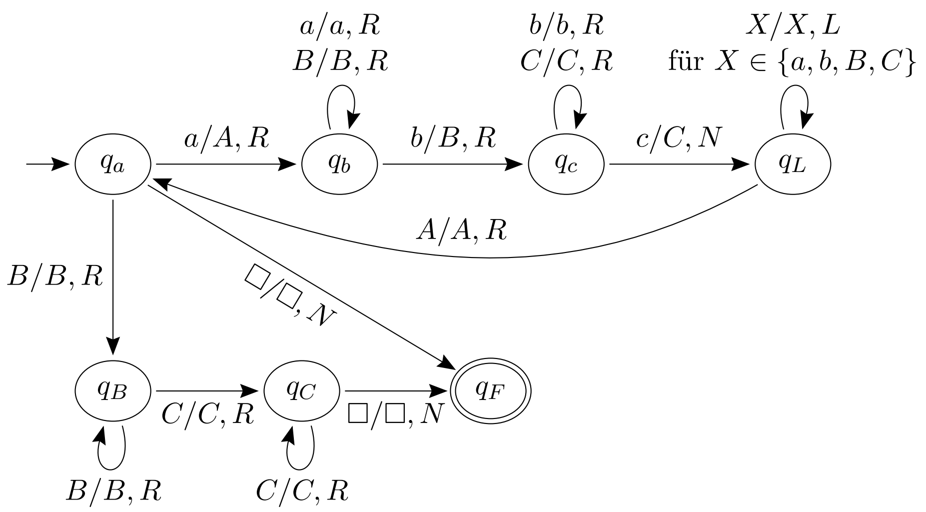
Sei $\Sigma = \{a, b, c\}$. Geben Sie TM M an, so dass:

$$L(M) = \{a^n b^n c^n \mid n \geq 0\}$$

Tipp: Es gibt verschiedene Webseiten auf denen Turing Maschinen interaktiv konstruiert und simuliert werden können, z.B. `https://wimmers.github.io/turing-machine-viz/` oder `https://turingmachinesimulator.com/`.

[1]Wenn Sie Automata Tutor noch nicht verwendet haben, folgen Sie erst den Schritten in Ü1.2, um sich richtig zu registrieren.

Lösungsskizze. Für diese Aufgabe gibt es eine Videolösung: Link
Idee: Ersetze für jedes a je ein a, b und c durch x, wobei nach dem Ersetzen die restlichen Buchstaben der gleichen Art unverändert gelassen werden. Wenn am Ende nur noch x auf dem Band stehen, terminiere. Sonst bleibt die Berechnung in einem Nichtendzustand stecken.
Wir schreiben $\square$ für eine leere Bandzelle.
Sei TM $M = (\{q_a, q_b, q_c, q_L, q_B, q_C, q_F\}, \{a, b, c\}, \Sigma \cup \{A, B, C, \square\}, \delta, q_a, \square, \{q_F\})$.



Übung und Nachbereitung

Übungsaufgabe Ü9.4. (*CFG* $\leftrightarrow$ *PDA*)

Wie Sie in der Vorlesung gesehen haben (ab Folie 203), können kontextfreie Grammatiken und Pushdown-Automaten sich gegenseitig simulieren. Wir üben nun diese Übersetzungen zwischen CFG und PDA.

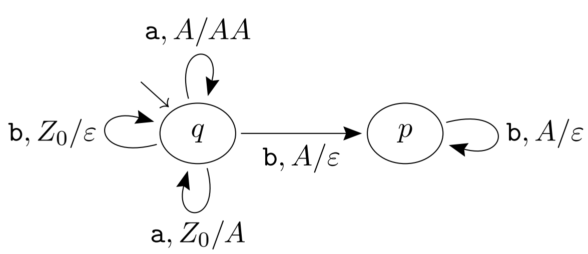
(a) Überführen Sie die folgende CFG $G = (\{S\}, \{a, b, c\}, P, S)$ mit Hilfe des Satzes der Vorlesung in einen PDA M mit $L_\varepsilon(M) = L(G)$:

$$S \to SS \mid aSb \mid bSa \mid c$$

(b) Übersetzen Sie den folgenden PDA $M = (\{q, p\}, \{a, b\}, \{Z_0, A, B\}, q, Z_0, \delta)$ in eine CFG G mit $L_\varepsilon(M) = L(G)$, wobei δ definiert ist durch:

$$\delta(q, a, A) = \{(q, AA)\} \qquad \delta(q, b, A) = \{(p, \epsilon)\} \qquad \delta(p, b, A) = \{(p, \epsilon)\}$$
$$\delta(q, a, Z_0) = \{(q, A)\} \qquad \delta(q, b, Z_0) = \{(q, \epsilon)\}$$

Geben Sie außerdem eine Ableitung in der konstruierten Grammatik für die Wörter b und $aabb$ an.



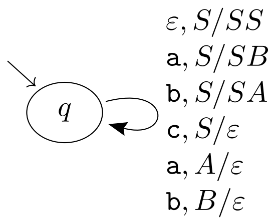
Lösungsskizze.

(a) Normalform für Übersetzung:

$$
\begin{aligned}
S &\to SS \mid aSB \mid bSA \mid c \\
A &\to a \\
B &\to b
\end{aligned}
$$

PDA $M = (\{q\}, \{a, b, c\}, \{S, A, B\}, q, S, \delta)$ mit δ:

$$
\begin{aligned}
qS &\xrightarrow{\varepsilon} qSS \\
qS &\xrightarrow{a} qSB \\
qS &\xrightarrow{b} qSA \\
qS &\xrightarrow{c} q\varepsilon \\
qA &\xrightarrow{a} q\varepsilon \\
qB &\xrightarrow{b} q\varepsilon
\end{aligned}
$$



(b) $G = ((\{p, q\} \times \{Z_0, A, B\} \times \{p, q\}) \cup \{S\}, \{a, b\}, P, S)$ mit den Produktionen P:

$$
\begin{array}{lcl}
S & \to & [q, Z_0, p] \mid [q, Z_0, q] \\
[q, Z_0, q] & \to & a[q, A, q] \\
[q, Z_0, p] & \to & a[q, A, p] \\
[q, A, q] & \to & a[q, A, q][q, A, q] \mid a[q, A, p][p, A, q] \\
[q, A, p] & \to & a[q, A, q][q, A, p] \mid a[q, A, p][p, A, p] \mid b \\
[p, A, p] & \to & b \\
[q, Z_0, q] & \to & b
\end{array}
$$

Ableitungen: $S \to [q, Z_0, q] \to b$ und $S \to [q, Z_0, p] \to a[q, A, p] \to aa[q, A, p][p, A, p] \to aab[p, A, p] \to aabb$.

Übungsaufgabe Ü9.5. (*TM Berechnung*)

(a) Konstruieren Sie eine Turingmaschine $T = (Q, \Sigma, \Gamma, q_0, \Box, \delta, F)$, mit $\Sigma = \{|\}$, die eine eingegebene Strichzahl verdoppelt.

(b) (Optional) In der Vorlesung haben Sie auf `https://turingmachinesimulator.com/`, Beispiel “Duplicate Binary String”, gesehen, wie eine Maschine für ein Wort $w \in \Sigma = \{0, 1\}$ das Wort ww^R erzeugt. Modifizieren Sie die Maschine auf der Website, sodass sie stattdessen das Wort ww erzeugt.

Lösungsskizze. Die Berechnung erfolgt, indem jeweils ein Strich am linken Anfang der Strichfolge durch eine Markierung x ersetzt wird und anschließend am rechten Ende der Strichfolge eine Markierung y angefügt wird. Falls keine Strichzeichen mehr vorhanden sind, werden alle Zeichen in Striche umgewandelt.

Sei $T = (Q, \Sigma, \Gamma, \delta, q_0, \Box, F)$ mit $Q = \{q_0, q_1, q_2, q_3, q_4, q_f\}$, $\Sigma = \{|\}$, $\Gamma = \{|, x, y, \Box\}$ und $F = \{q_f\}$.

Die Zustände und Zeichenmengen kann man der folgenden Tabelle der Übergangsfunktion δ entnehmen.

Übergang			Kommentar
$\delta(q_0, \Box)$	$\to$	$(q_3, \Box, L)$	Nichts mehr zu verdoppeln
$\delta(q_0, \vert)$	$\to$	(q_1, x, R)	$\vert$ wird verdoppelt, x ist ein Hilfszeichen
$\delta(q_0, y)$	$\to$	(q_0, y, R)	Überspringe Hilfszeichen y
$\delta(q_1, \Box)$	$\to$	(q_2, y, N)	Schreibe $\vert$ an rechten Rand, markiert durch Hilfszeichen y
$\delta(q_1, \vert)$	$\to$	$(q_1, \vert, R)$	An den rechten Rand gehen
$\delta(q_1, y)$	$\to$	(q_1, y, R)	An den rechten Rand gehen
$\delta(q_2, \vert)$	$\to$	$(q_2, \vert, L)$	Zurück zum nächsten $\vert$
$\delta(q_2, y)$	$\to$	(q_2, y, L)	Zurück zum nächsten $\vert$
$\delta(q_2, x)$	$\to$	(q_0, x, R)	Nächstes $\vert$ verdoppeln oder halten
$\delta(q_3, y)$	$\to$	$(q_3, \vert, L)$	Hilfszeichen durch $\vert$ ersetzen
$\delta(q_3, x)$	$\to$	$(q_3, \vert, L)$	Hilfszeichen durch $\vert$ ersetzen
$\delta(q_3, \Box)$	$\to$	$(q_f, \Box, R)$	Halten

Hier der Code für https://turingmachinesimulator.com/:

```
name: Duplicate unary string
init: q0
accept: qf

q0,_
q3,_,<

q0,1
q1,x,>

q0,y
q0,y,>

q1,_
q2,y,-

q1,1
q1,1,>

q1,y
q1,y,>

q2,1
q2,1,<

q2,y
q2,y,<

q2,x
```

q0 , x, >

q3 , y
q3 ,1 , <

q3 , x
q3 ,1 , <

q3 ,_

qf ,_,>

In der Vorlesung wurde die Annahme gemacht, dass die Übergangsfunktion δ einer Turingmaschine folgende Eigenschaft erfüllt:

$$\delta(q, \mathsf{a}) \textit{ ist nicht definiert für alle } q \in \mathsf{F},\ \mathsf{a} \in \Gamma.$$

Sei $\mathcal{M}_A$ die Menge der Turingmaschinen, die diese Annahme erfüllen, und sei $\mathcal{M}$ die Menge aller Turingmaschinen. Es gilt somit $\mathcal{M}_A \subsetneq \mathcal{M}$.

Für $\mathsf{M} \in \mathcal{M}$ mit $\mathsf{M} = (\mathsf{Q}, \Sigma, \Gamma, \delta, \mathsf{q}_0, \square, \mathsf{F})$ definiere:

- $\mathsf{L}_F(\mathsf{M}) = \{w \in \Sigma^* \mid \exists \alpha, \beta \in \Gamma^*, f \in \mathsf{F}.\ (\varepsilon, \mathsf{q}_0, w) \rightarrow^*{}_{\mathsf{M}} (\alpha, f, \beta)\}$.
  (Menge der Wörter, für die die Maschine einen Endzustand irgendwann besucht.)
- $\mathsf{L}_H(\mathsf{M}) = \{w \in \Sigma^* \mid \exists \alpha, \beta \in \Gamma^*, q \in \mathsf{Q}.\ (\varepsilon, \mathsf{q}_0, w) \rightarrow^*_M (\alpha, q, \beta)$ und $\delta(q, \mathit{first}(\beta))$ ist nicht definiert$\}$.
  (Menge der Wörter, für die die Maschine hält.)

Übungsaufgabe Ü9.6. (*TM Akzeptanzbedingungen*)

. Begründen Sie folgende Aussagen, indem Sie eine passende Konstruktion angeben.

(a) Für jede Turing-Maschine $\mathsf{M} \in \mathcal{M}_A$ gibt es eine Turing-Maschine $\mathsf{M}' \in \mathcal{M}$ mit $\mathsf{L}_F(\mathsf{M}) = \mathsf{L}_H(\mathsf{M}')$.

(b) Für jede Turing-Maschine $\mathsf{M} \in \mathcal{M}$ gibt es eine Turing-Maschine $\mathsf{M}' \in \mathcal{M}_A$ mit $\mathsf{L}_H(\mathsf{M}) = \mathsf{L}_F(\mathsf{M}')$.

(c) Für jede Turing-Maschine $\mathsf{M} \in \mathcal{M}_A$ gibt es eine Turing-Maschine $\mathsf{M}' \in \mathcal{M}$ mit $\mathsf{L}_F(\mathsf{M}) = \mathsf{L}_F(\mathsf{M}')$.

(d) Für jede Turing-Maschine $\mathsf{M} \in \mathcal{M}$ gibt es eine Turing-Maschine $\mathsf{M}' \in \mathcal{M}_A$ mit $\mathsf{L}_F(\mathsf{M}) = \mathsf{L}_F(\mathsf{M}')$.

(e) Für jede Turing-Maschine $\mathsf{M} \in \mathcal{M}$ gibt es eine Turing-Maschine $\mathsf{M}' \in \mathcal{M}$ mit $\mathsf{L}_F(\mathsf{M}) = \mathsf{L}_H(\mathsf{M}')$.

Lösungsskizze.

(a) Füge einen neuen Fangzustand q_t ein und mache die Transitionsfunktion total für Nichtendzustände. Formal:

$$\mathsf{Q}' = \mathsf{Q} \cup \{q_t\} \qquad \begin{aligned} \delta' =& \delta \cup \{(q_t, a, q_t, a, N) \mid a \in \Gamma\} \cup \\ & \{(q, a, q_t, a, N) \mid a \in \Gamma \wedge q \in \mathsf{Q} \setminus \mathsf{F} \wedge \delta(q, a) \text{ ist undefiniert}\} \end{aligned}$$

(b) Füge einen neuen Endzustand q_f ein und mache die Transitionsfunktion total (außer für q_f). Formal:

$$\mathsf{Q}' = \mathsf{Q} \cup \{q_f\} \qquad \mathsf{F}' = \{q_f\} \qquad \delta' = \delta \cup \{(q, a, q_f, a, N) \mid a \in \Gamma \wedge q \in \mathsf{Q} \wedge \delta(q, a) \text{ ist undefiniert}\}$$

(c) Folgt sofort, da $\mathcal{M}_A \subsetneq \mathcal{M}$.

(d) Lösche alle ausgehenden Kanten von Endzuständen. Formal:

$$\delta' = \delta \setminus \mathsf{F} \times \Gamma \times \mathsf{Q} \times \Gamma \times \{L, R, N\}$$

(e) Lösche alle ausgehenden Kanten von Endzuständen und mache die Transitionsfunktion total für Nichtendzustände. Formal:

$$\begin{aligned} \mathsf{Q}' =& \mathsf{Q} \cup \{q_t\} \\ \delta' =& (\delta \cup \{(q_t, a, q_t, a, N) \mid a \in \Gamma\} \cup \\ & \{(q, a, q_t, a, N) \mid a \in \Gamma \wedge q \in \mathsf{Q} \setminus \mathsf{F} \wedge \delta(q, a) \text{ ist undefiniert}\}) \setminus \mathsf{F} \times \Gamma \times \mathsf{Q} \times \Gamma \times \{L, R, N\} \end{aligned}$$