Universität Siegen
Lehrstuhl Theoretische Informatik
Markus Lohrey

Übungsblatt 14

Aufgabe 1. Wahr oder falsch?

(a) Jeder linear beschränkte Automat ist eine Turingmaschine.

(b) Eine Turingmaschine darf nie das Blanksymbol $\Box$ auf das Band schreiben.

(c) Typ-0-Sprachen sind unter Homomorphismenbildung abgeschlossen.

Lösung zu Aufgabe 1.

(a) wahr: Ein LBA ist eine Turingmaschine mit Einschränkungen.

(b) falsch: Nur LBAs dürfen $\Box$ nicht schreiben.

(c) wahr: Einfach den Homomorphismus auf die Typ-0-Grammatik anwenden und man erhält wieder eine Typ-0-Grammatik.

Aufgabe 2. Sei $M = (\{z_0, z_e\}, \{a, b\}, \{a, b, \Box\}, \delta, z_0, \Box, \{z_e\})$ eine Turingmaschine, wobei δ gegeben ist durch:

$$\begin{aligned} \delta(z_0, a) &= (z_e, a, R) \\ \delta(z_0, b) &= (z_0, b, R) \\ \delta(z_0, \Box) &= (z_0, \Box, N) \end{aligned}$$

Bei Eingabe welcher Wörter $w \in \{a, b\}^*$ gelangt M in einen Endzustand?

Lösung zu Aufgabe 2. M durchläuft das Band von links nach rechts und geht in den Endzustand z_e über, sobald ein a gelesen wurde.

Die Menge aller Wörter, mit denen M in einen Endzustand gelangt, ist daher die Menge aller Wörter, die mindestens ein a enthalten.
Dementsprechend ist die gesuchte Menge

$$L((a \mid b)^* a(a \mid b)^*) = L(b^* a(a \mid b)^*) = \{b^n ax \mid n \geq 0, x \in \{a, b\}^*\}.$$

Aufgabe 3. Geben Sie eine Turingmaschine an, die bei Eingabe eines Wortes $w \in \{a, b, c\}^*$ genau dann in einen Endzustand gelangt, wenn

$$w \in \{a^n b^n c^n \mid n \in \mathbb{N}\}.$$

Lösung zu Aufgabe 3. Die Grundidee ist, dass wiederholt zunächst ein a, dann ein b und dann ein c auf dem Band weggestrichen wird, wobei „wegstreichen“ in diesem Fall bedeutet, dass wir a durch $\#_a$ ersetzen, b durch $\#_b$ und c durch $\#_c$.
Das Wort wird nur dann akzeptiert, wenn schließlich keine a, b, c mehr auf dem Band stehen.

- $M = (Z, \Sigma, \Gamma, \delta, z_0, \square, \{z_e\})$
- $Z = \{z_0, z_1, z_2, z_3, z_4, z_e\}$
- $\Sigma = \{a, b, c\}$
- $\Gamma = \Sigma \cup \{\square, \#_a, \#_b, \#_c\}$

Dabei ist δ wie folgt definiert, Übergänge die zu Endlosschleifen führen (das Wort also nicht akzeptieren) sind **fettgedruckt** markiert.

$$\delta(z_0, \square) = (z_e, \square, N) \tag{1}$$
$$\delta(z_0, a) = (z_1, \#_a, R) \tag{1}$$
$$\boldsymbol{\delta(z_0, b) = (z_0, b, N)} \tag{1}$$
$$\boldsymbol{\delta(z_0, c) = (z_0, c, N)} \tag{1}$$
$$\delta(z_0, \#_b) = (z_4, \#_b, R) \tag{1}$$

$$\delta(z_1, a) = (z_1, a, R) \tag{2}$$
$$\delta(z_1, \#_b) = (z_1, \#_b, R) \tag{2}$$
$$\delta(z_1, b) = (z_2, \#_b, R) \tag{2}$$
$$\boldsymbol{\delta(z_1, c) = (z_1, c, N)} \tag{2}$$
$$\boldsymbol{\delta(z_1, \#_c) = (z_1, \#_c, N)} \tag{2}$$
$$\boldsymbol{\delta(z_1, \square) = (z_1, \square, N)} \tag{2}$$

$$\delta(z_2, b) = (z_2, b, R) \tag{3}$$
$$\delta(z_2, \#_c) = (z_2, \#_c, R) \tag{3}$$
$$\delta(z_2, c) = (z_3, \#_c, L) \tag{3}$$
$$\boldsymbol{\delta(z_2, a) = (z_2, a, N)} \tag{3}$$
$$\boldsymbol{\delta(z_2, \square) = (z_2, \square, N)} \tag{3}$$

$$\delta(z_3, x) = (z_3, x, L) \qquad \text{für alle } x \in \{a, b, c, \#_b, \#_c\} \tag{4}$$
$$\delta(z_3, \#_a) = (z_0, \#_a, R) \tag{4}$$

$$\delta(z_4, \#_b) = (z_4, \#_b, R) \tag{5}$$
$$\delta(z_4, \#_c) = (z_4, \#_c, R) \tag{5}$$
$$\delta(z_4, \square) = (z_e, \square, N) \tag{5}$$
$$\boldsymbol{\delta(z_4, a) = (z_4, a, N)} \tag{5}$$
$$\boldsymbol{\delta(z_4, b) = (z_4, b, N)} \tag{5}$$
$$\boldsymbol{\delta(z_4, c) = (z_4, c, N)} \tag{5}$$

(1) Ersetzen von a durch $\#_a$ (Zustand z_0)

Das leere Wort (die TM beginnt mit dem Lesekopf auf einem $\square$) wird direkt akzeptiert. Ein a ersetzen wir durch $\#_a$ und beginnen mit z_1 die Suche nach dem dazugehörgen b. Steht der Lesekopf auf einem b oder c kann das Wort nicht gültig sein und wir gehen in eine Endlosschleife über. Beachten Sie, dass es auch (in einer nichtdeterministische Turingmaschine) möglich wäre, einfach keine Transitionen für $\delta(z_0, b)$ und $\delta(z_0, c)$ anzugeben, da auch in diesem Fall eine Sackgasse in der Berechnung erreicht wird und niemals ein Endzustand erreicht würde. Steht der Lesekopf im Zustand z_0 auf einem $\#_b$ (dies kann nur passieren, nachdem wir in einem Durchlauf ein a, ein b und ein c ersetzt haben und zurück zum ersten Symbol das auf ein $\#_a$' folgt gelaufen sind) haben wir alle a's am Anfang des Wortes „verarbeitet“ und müssen noch mit z_4 prüfen, ob auf dem Rest des Bandes noch b's oder c's übrig sind.

(2): Ersetzen von b durch $\#_b$ (Zustand z_1)

Alle a's und $\#_b$'s (also bereits ersetzte b's) werden übersprungen. Das erste noch nicht verarbeitete b wird durch $\#_b$ ersetzt und wir beginnen mit z_2 die Suche nach dem dazugehörigen c. Falls wir kein b finden, kann das Wort nicht gültig sein und wir gehen in eine Endlosschleife über.

(3): Ersetzen von c durch $\#_c$ (Zustand z_2)

Alle b's und $\#_c$'s (also bereits ersetzte c's) werden übersprungen. Das erste noch nicht verarbeitete c wird durch ein $\#_c$ ersetzt, wodurch ein Ersetzungsdurchgang (von einem a, einem b und einem c) beendet wurde. Falls wir kein c finden kann das Wort nicht gültig sein und wir gehen in eine Endlosschleife über.

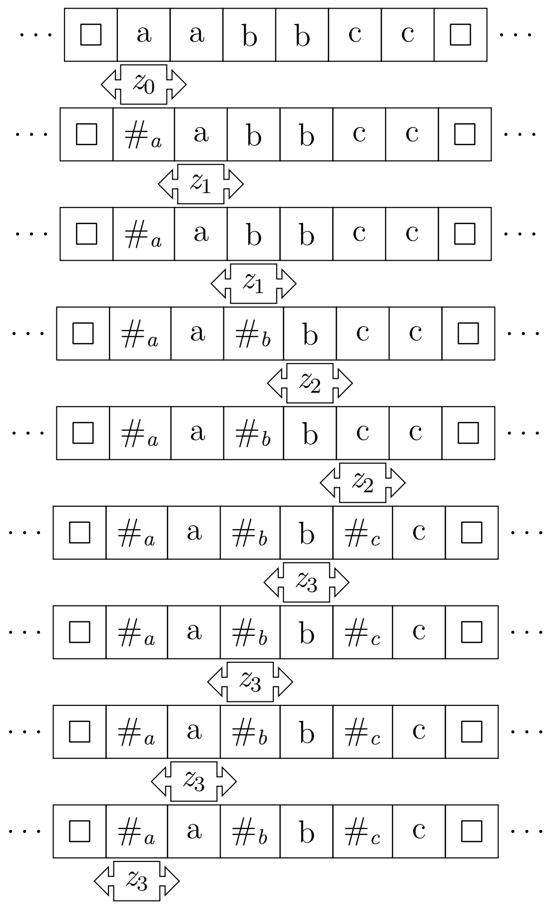
(4): Zurück zum letzten $\#_a$ (Zustand z_3) Wir gehen mit z_3 zum letzten $\#_a$ zurück und beginnen die Prozedur anschließend von vorne.

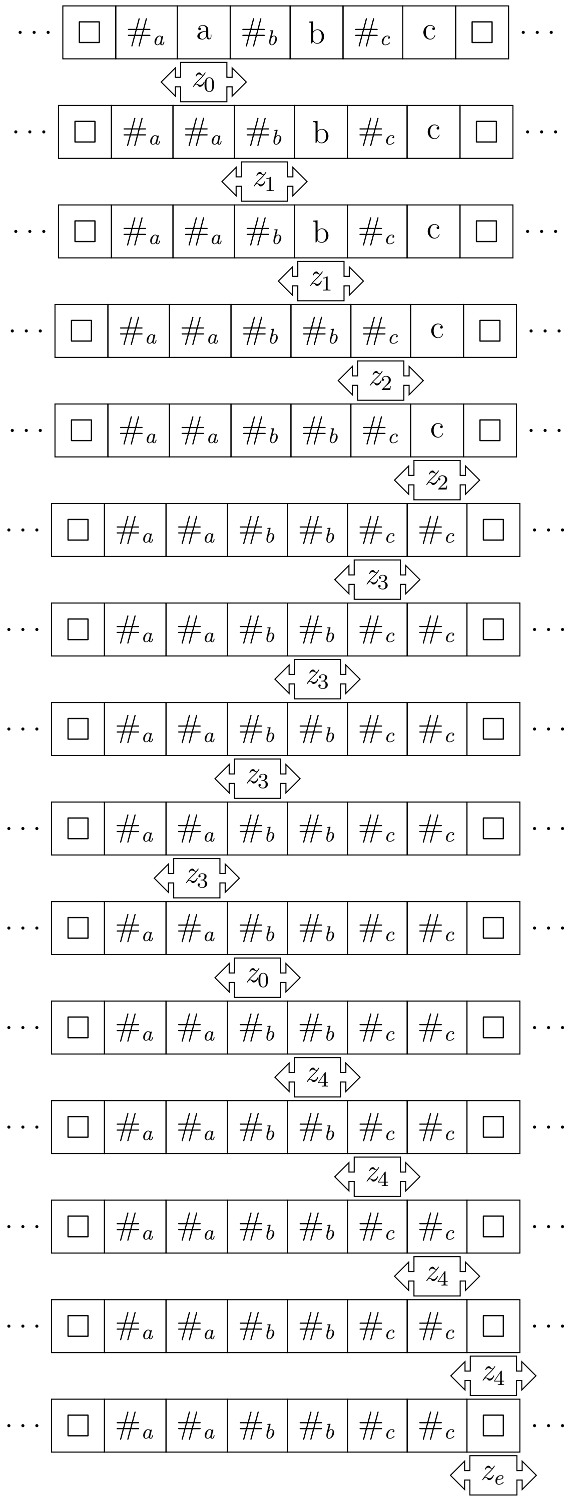
(5): Prüfen, ob alle Symbole ersetzt wurden (Zustand z_4)

Wir durchlaufen das Wort von links nach rechts. Falls noch b's oder c's auf dem Band stehen, wird das Wort nicht akzeptiert. So wird der Fall $w = a^k b^l c^m$ mit $l > k \vee m > k$ ausgeschlossen.

Beachte: Damit die Turingmaschine deterministisch ist, müssten wir genau genommen noch für alle Paare $(z, x) \in (Z \setminus \{z_e\}) \times \Gamma$, für die δ noch nicht definiert ist, $\delta(z, x)$ angeben (allerdings wird die Turingmaschine diese Transitionen nie verwenden, d.h. es ist egal, wie wir dort $\delta(z, x)$ definieren). Alternativ können wir die Turingmaschine als nichtdeterministische Turingmaschine auffassen (dann ist $\delta(z, x)$ formal gesehen eine Menge, d.h. wir müssten in der obigen Definition von δ noch Mengenklammern auf den rechten Seiten ergänzen).

Durchlauf der TM auf der Eingabe *aabbcc*





$\cdots$ □ $\#_a$ a $\#_b$ b $\#_c$ c □ $\cdots$
z_0
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ b $\#_c$ c □ $\cdots$
z_1
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ b $\#_c$ c □ $\cdots$
z_1
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ c □ $\cdots$
z_2
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ c □ $\cdots$
z_2
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ $\#_c$ □ $\cdots$
z_3
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ $\#_c$ □ $\cdots$
z_3
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ $\#_c$ □ $\cdots$
z_3
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ $\#_c$ □ $\cdots$
z_3
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ $\#_c$ □ $\cdots$
z_0
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ $\#_c$ □ $\cdots$
z_4
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ $\#_c$ □ $\cdots$
z_4
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ $\#_c$ □ $\cdots$
z_4
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ $\#_c$ □ $\cdots$
z_4
$\cdots$ □ $\#_a$ $\#_a$ $\#_b$ $\#_b$ $\#_c$ $\#_c$ □ $\cdots$
z_e

Aufgabe 4. Geben Sie eine Typ-1-Grammatik an, die die Sprache

$$L = \{w \in \{a, b, c\}^* \mid \#_a(w) = \#_b(w) = \#_c(w)\}$$

erzeugt. Zur Erinnerung: $\#_x(w)$ ist die Anzahl der Zeichen $x \in \Sigma$ in w.

Lösung zu Aufgabe 4. Es gilt $L = L(G)$ mit $G = (V, \Sigma, P, S)$ und $V = \{S, A, B, C\}$ und

$$\begin{aligned} P =& \{S \to \varepsilon \mid SS \mid ABC, AB \to BA, AC \to CA, BA \to AB, BC \to CB, \\ & CA \to AC, CB \to BC, A \to a, B \to b, C \to c\}. \end{aligned}$$

Erklärung: Durch S produzieren wir beliebig viele ABC-Blöcke. Durch Produktionen der Form $DE \to ED$ können wir die Zeichen frei vertauschen. Schließlich können wir mit den letzten 3 Produktionen auch die Nichtterminale in Terminalsymbole umwandeln.

Aufgabe 5. Geben Sie eine Turingmaschine an, die bei Eingabe eines Wortes $w \in \{a, b\}^*$ das Wort w^r auf das Band schreibt, den Kopf auf das erste Symbol von w^r bewegt und in einen Endzustand übergeht.

Lösung zu Aufgabe 5. Die Idee ist, zuerst rechts neben das Wortende das Symbol \$ auf das Band zu schreiben, und anschließend das Eingabewort an dem Symbol \$ nach rechts zu spiegeln.

- $M = (Z, \Sigma, \Gamma, \delta, z_0, \Box, \{z_e\})$
- $Z = \{z_0, z_1, z_2, z_3, z_4, z_5, z_e\}$
- $\Sigma = \{a, b\}$
- $\Gamma = \Sigma \cup \{\Box, \#, \$\}$

Dabei ist δ wie folgt definiert:

<u>z_0: Anhängen von \$ an das Ende des Wortes</u>

- $\delta(z_0, a) = (z_0, a, R)$
- $\delta(z_0, b) = (z_0, b, R)$
- $\delta(z_0, \Box) = (z_1, \$, L)$

<u>z_1: Suchen und merken des nächsten Buchstabens, der an \$ gespiegelt wird</u>

- $\delta(z_1, a) = (z_2, \#, R)$

- $\delta(z_1, b) = (z_3, \#, R)$
- $\delta(z_1, \#) = (z_1, \#, L)$
- $\delta(z_1, \square) = (z_5, \square, R)$

Durch den Zustand z_2 merkt man sich, dass ein a gefunden wurde und durch z_3 merkt man sich, dass ein b gefunden wurde. In beiden Fällen wird der gelesene Buchstabe durch # ersetzt, so dass man in den nächsten Runden alle bereits durch # ersetzten Buchstaben überspringen kann. Wenn $\square$ erreicht wird, so wurden alle Buchstaben von w bereits gespiegelt und man wechselt in den Zustand z_5, in dem alle #'s und \$ durch $\square$ ersetzt werden.

z_2: Anhängen von a am rechten Ende des gespiegelten Wortes

- $\delta(z_2, x) = (z_2, x, R)$ für alle $x \in \{\#, \$, a, b\}$
- $\delta(z_2, \square) = (z_4, a, L)$

z_3: Anhängen von b am rechten Ende des gespiegelten Wortes

- $\delta(z_3, x) = (z_3, x, R)$ für alle $x \in \{\#, \$, a, b\}$
- $\delta(z_3, \square) = (z_4, b, L)$

z_4: Lesekopf zurück (nach links) zum \$-Symbol bewegen

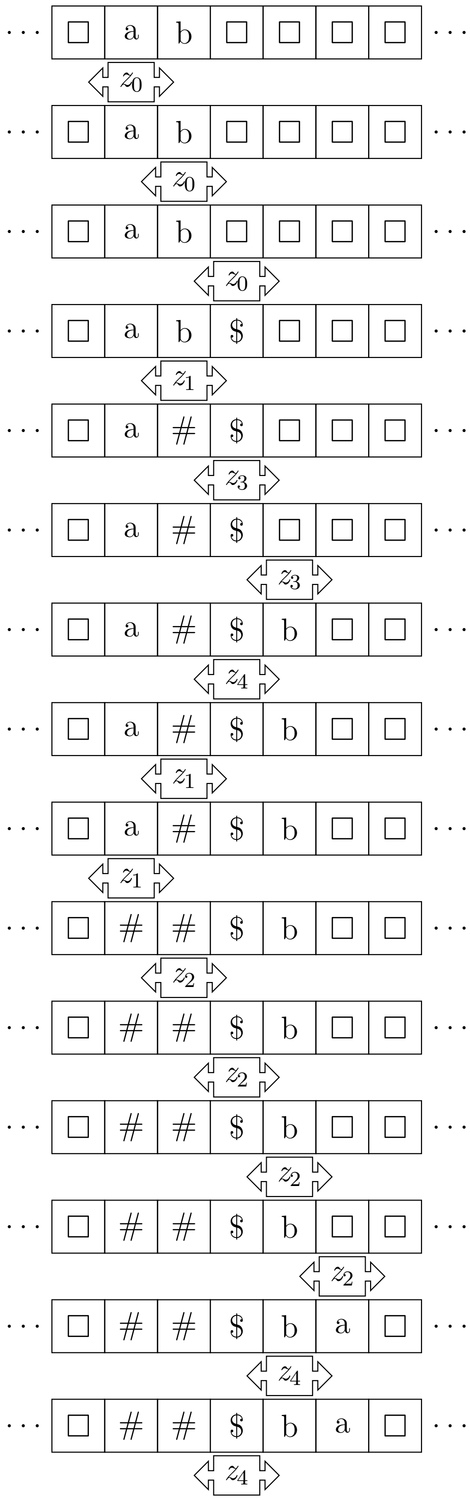
- $\delta(z_4, a) = (z_4, a, L)$
- $\delta(z_4, b) = (z_4, b, L)$
- $\delta(z_4, \$) = (z_1, \$, L)$

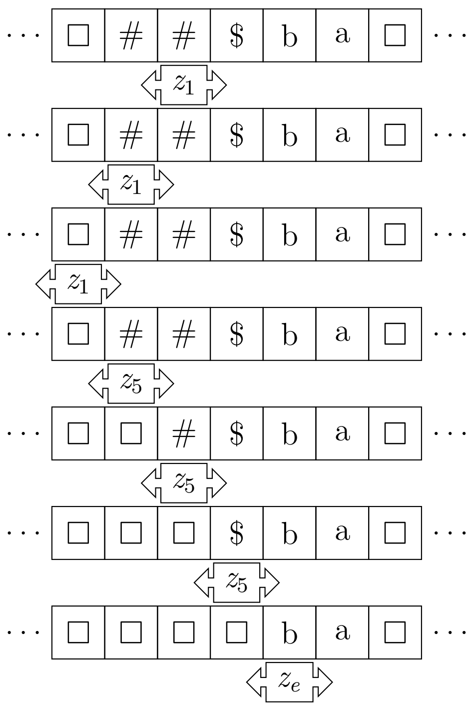
z_5: Alle #'s und das \$-Symbol durch $\square$'s überschreiben

- $\delta(z_5, \#) = (z_5, \square, R)$
- $\delta(z_5, \$) = (z_e, \square, R)$

Auch hier gilt wieder die Bemerkung aus Aufgabe 3: Damit die Turingmaschine deterministisch ist, müssten wir formal gesehen noch alle weiteren Werte $\delta(z, x)$ definieren, auch wenn diese nie verwendet werden. Ein Durchlauf dieser Turingmaschine sieht wie folgt aus:

Durchlauf der TM für die Eingabe ab





Aufgabe 6. Sei $M = (Z, \Sigma, \delta, z_0, E)$ ein deterministischer endlicher Automat. Geben Sie eine Turingmaschine an, die bei Eingabe eines Wortes $w \in \Sigma^*$ genau dann in einen Endzustand gelangt, wenn $w \in T(M)$.

Lösung zu Aufgabe 6.

- $M' = (Z', \Sigma, \Gamma', \delta', z_0, \Box, \{z_e\})$
- $Z' = Z \cup \{z_e\}$
- $\Gamma' = \Sigma \cup \{\Box\}$

Definiere für alle $a \in \Sigma$ und für alle $z \in Z$

$$\delta'(z, a) = (\delta(z, a), a, R)$$

$$\delta'(z, \Box) = \begin{cases} (z_e, \Box, N) & \text{falls } z \in E \\ (z, \Box, N) & \text{sonst} \end{cases}$$

<u>Beachte</u>: Wir können nicht E als Menge der Endzustände von M' verwenden, da wir sicherstellen müssen, dass das ganze Wort gelesen wurde.