Lösungen zur GTI-Klausur vom 16.09.2013

Aufgabe 1

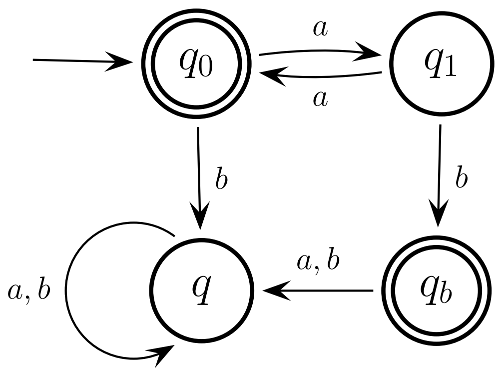
Sei $L = \{a^n b^{n \,\mathbf{mod}\, 2} \mid n \in \mathbb{N}\} = \{a^{2n} \mid n \in \mathbb{N}\} \cup \{a^{2n+1}b \mid n \in \mathbb{N}\}$.

(a) Man wähle $\alpha = (aa)^\star \mid (aa)^\star ab$.

(b) Ein DEA, der L akzeptiert, sollte stets unterscheiden, ob er bisher eine gerade oder eine ungerade Anzahl von as gelesen hat und in Abhängigkeit davon entweder kein b oder genau ein b am Ende des Wortes zulassen. Dem entsprechend wählt man $A = (\Sigma, Q, s, F, \delta)$ mit $\Sigma = \{a, b\}$, $Q = \{q_0, q_1, q_b, q\}$, $s = q_0$, $F = \{q_0, q_b\}$ und $\delta : Q \times \Sigma \to Q$ definiert durch

δ	a	b
q_0	q_1	q
q_1	q_0	q_b
q_b	q	q
q	q	q

Graphisch:



(c) Zum Beweis von $L = L(A)$ genügt es zu zeigen, dass für alle $w \in \Sigma^*$

$$(q_0, w) \;\vdash_A^* \begin{cases} (q_0, \varepsilon) & \text{falls } \exists\, n \in \mathbb{N}.\ w = a^{2n} \\ (q_1, \varepsilon) & \text{falls } \exists\, n \in \mathbb{N}.\ w = a^{2n+1} \\ (q_b, \varepsilon) & \text{falls } \exists\, n \in \mathbb{N}.\ w = a^{2n+1}b \\ (q, \varepsilon) & \text{sonst} \end{cases} \qquad (*)$$

Aus (*) erhält man nämlich

$$\begin{aligned} w \in L \;&\Leftrightarrow\; \exists\, n \in \mathbb{N}.\ w = a^{2n} \vee w = a^{2n+1}b && \text{per Definition von } L \\ &\Leftrightarrow\; \exists\, f \in \{q_0, q_b\}.\ (q_0, w) \vdash_A^* (f, \varepsilon) && \text{wegen } (*) \\ &\Leftrightarrow\; w \in L(A) && \text{wegen } F = \{q_0, q_b\} \end{aligned}$$

Den Beweis von (*) führt man durch Induktion über $|w|$.

$|w| = 0$, d.h. $w = \varepsilon$:
Wegen $w = a^{2\cdot 0}$ und $(q_0, w) \vdash_A^* (q_0, \varepsilon)$ ist (*) erfüllt.

$|w| > 0$:
Im Falle $w = a^n = a^{n-1}a$ gilt: Falls n gerade ist, dann ist $n-1$ ungerade, also nach Induktionsannahme $(q_0, a^{n-1}) \vdash_A^* (q_1, \varepsilon)$ und damit $(q_0, w) \vdash_A^* (q_1, a) \vdash_A (q_0, \varepsilon)$. Falls n ungerade ist, dann ist $n-1$ gerade, also nach Induktionsannahme $(q_0, a^{n-1}) \vdash_A^* (q_0, \varepsilon)$ und damit $(q_0, w) \vdash_A^* (q_0, a) \vdash_A (q_1, \varepsilon)$.

Im Falle $w = a^n b$ gilt: Falls n gerade ist, dann gilt nach Induktionsannahme $(q_0, a^n) \vdash_A^* (q_0, \varepsilon)$ und damit $(q_0, w) \vdash_A^* (q_0, b) \vdash_A (q, \varepsilon)$. Falls n ungerade ist, dann gilt nach Induktionsannahme $(q_0, a^n) \vdash_A^* (q_1, \varepsilon)$ und damit $(q_0, w) \vdash_A^* (q_1, b) \vdash_A (q_b, \varepsilon)$.

In allen anderen Fällen gilt $w = vz$ mit $v \notin \{a\}^*$ und $z \in \{a, b\}$. Dann gilt nach Induktionsannahme $(q_0, v) \vdash_A^* (q_b, \varepsilon)$ oder $(q_0, v) \vdash_A^* (q, \varepsilon)$ und damit in beiden Fällen $(q_0, w) \vdash_A^* (q, \varepsilon)$ weil man sowohl von q_b als auch von q nur nach q gelangen kann.

Damit ist (*) in allen Fällen erfüllt.

(d) Nach dem Verfahren aus der Vorlesung erhält man aus dem DEA in (c) die rechtslineare Grammatik $G = (\Sigma, N, S, P)$ mit $\Sigma = \{a, b\}$, $N = \{q_0, q_1, q_b, q\}$, $S = q_0$ und

$$P = \{ \begin{array}{rcl} q_0 & \to & aq_1 \mid bq \mid \varepsilon, \\ q_1 & \to & aq_0 \mid bq_b, \\ q_b & \to & aq \mid bq \mid \varepsilon, \\ q & \to & aq \mid bq \; \} \end{array}$$

Aufgabe 2

Sei $L = \{waw^R \mid w \in \{a, b\}^*\}$.

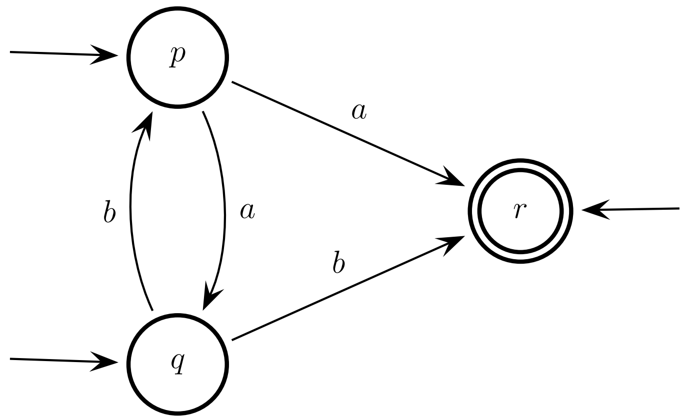
a. Nach dem starken Pumping Lemma genügt es zu zeigen: Für jedes $n \geq 1$ existiert ein $x \in L$ mit $|x| \geq n$ so, dass für jede Zerlegung $x = uvw$ mit $v \neq \varepsilon$ und $|uv| \leq n$ ein Pumpfaktor $i \in \mathbb{N}$ existiert mit $uv^iw \notin L$.

Sei $n \geq 1$. Man wähle $x = b^n ab^n$. Dann ist $x \in L$ und $|x| = 2n + 1 \geq n$. Sei $x = uvw$ mit $v \neq \varepsilon$ und $|uv| \leq n$. Dann besteht v nur aus bs, d.h. $v = b^k$ für ein $k \geq 1$ und es folgt $uv^2w = b^{n+k}ab^n \notin L$.

b. Nach dem Satz von Myhill-Nerode genügt es, unendlich viele Wörter in Σ^* zu finden, die paarweise L-unterscheidbar sind. Man wähle die unendlich vielen Wörter $b^n a$ mit $n \in \mathbb{N}$. Für alle $m \neq n$ gilt $b^n ab^n \in L$ und $b^m ab^n \notin L$, also sind $b^n a$ und $b^m a$ L-unterscheidbar.

Aufgabe 3

Sei $A = (\Sigma, Q, S, F, \Delta)$ der NDEA aus der Aufgabenstellung:



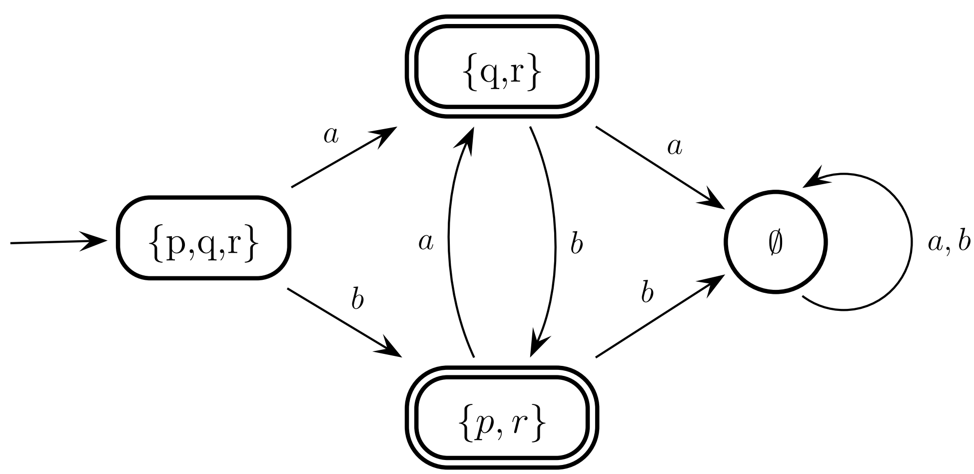
Der Potenzautomat $A' = (\Sigma, Q', s', F', \delta)$ zu A ist definiert durch

- $Q' = \wp(Q)$
- $s' = S = \{p, q, r\}$
- $F' = \{P \subseteq Q \mid P \cap F \neq \emptyset\} = \{P \subseteq Q \mid r \in P\}$
- $\delta : Q' \times \Sigma \to Q'$
  $\delta(P, z) = \{q \in Q \mid \exists p \in P.\ (p, z, q) \in \Delta\}$

Den erreichbaren Teil von A' erhält man also wie folgt:

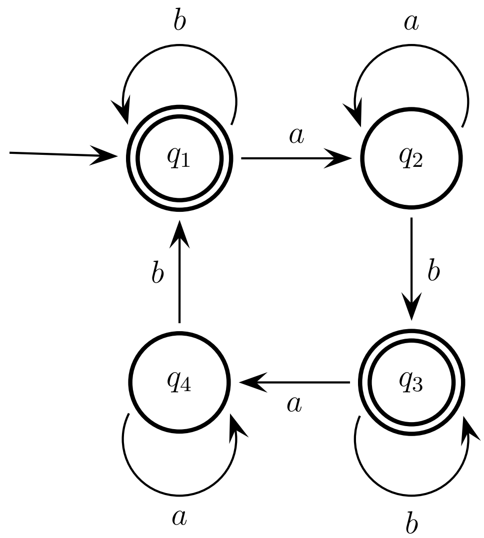
- $\delta(s', a) = \delta(\{p, q, r\}, a) = \{q, r\}, \quad \delta(s', b) = \delta(\{p, q, r\}, b) = \{p, r\}$
- $\delta(\{p, r\}, a) = \{q, r\}, \quad \delta(\{p, r\}, b) = \emptyset$
- $\delta(\{q, r\}, a) = \emptyset, \quad \delta(\{q, r\}, b) = \{p, r\}$
- $\delta(\emptyset, a) = \emptyset, \quad \delta(\emptyset, b) = \emptyset$

wobei $\{p, q, r\}$, $\{p, r\}$ und $\{q, r\}$ die erreichbaren Endzustände sind:



Aufgabe 4

Sei A der DEA



Um A zu minimieren, berechnet man die Äquivalenzrelation $\sim_A = \bigcap_{n\in\mathbb{N}} \sim_n$.

$\sim_0$: hat die beiden Äquivalenzklassen $F = \{q_1, q_3\}$ und $Q \setminus F = \{q_2, q_4\}$.

$\sim_1$: Wegen $\delta(q_1, a) = q_2 \sim_0 q_4 = \delta(q_3, a)$ und $\delta(q_1, b) = q_1 \sim_0 q_3 = \delta(q_3, b)$ gilt $q_1 \sim_1 q_3$. Wegen $\delta(q_2, a) = q_2 \sim_0 q_4 = \delta(q_4, a)$ und $\delta(q_2, b) = q_3 \sim_0 q_1 = \delta(q_4, b)$ gilt $q_2 \sim_1 q_4$. Also gilt $\sim_0 = \sim_1$ und damit auch $\sim_0 = \sim_A$.

Daraus erhält man den minimalen DEA $\bar{A} = (\Sigma, \bar{Q}, \bar{s}, \bar{F}, \bar{\delta})$ wie folgt:

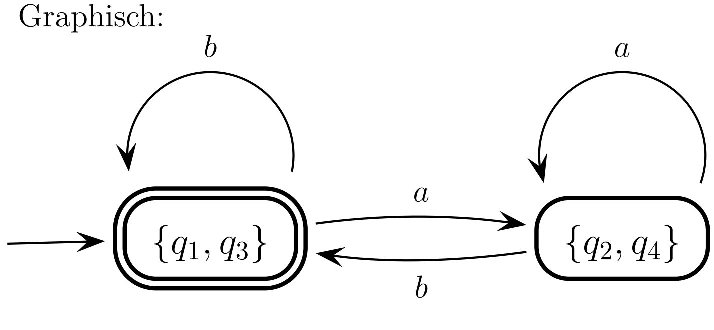
$\bar{Q}$ besteht aus den Äquivalenzklassen von $\sim_A$, also $\bar{Q} = \{F, Q \setminus F\} = \{\{q_1, q_3\}, \{q_2, q_4\}\}$, $\bar{s}$ ist die Äquivalenzklasse des Startzustands von A, also $\bar{s} = \{q_1, q_3\}$, $\bar{F}$ besteht aus den Äquivalenzklassen der Endzustände von Q, also $\bar{F} = \{\{q_1, q_3\}\}$ und $\bar{\delta}$ erhält man nach der Vorschrift

$$\bar{\delta}([q]_A, z) = [\delta(q, z)]_A \text{ für alle } q \in Q, z \in \Sigma$$

also

- $\bar{\delta}(\{q_1, q_3\}, a) = [\delta(q_1, a)]_A = [q_2]_A = \{q_2, q_4\}$
- $\bar{\delta}(\{q_1, q_3\}, b) = [\delta(q_1, b)]_A = [q_1]_A = \{q_1, q_3\}$
- $\bar{\delta}(\{q_2, q_4\}, a) = [\delta(q_2, a)]_A = [q_2]_A = \{q_2, q_4\}$
- $\bar{\delta}(\{q_2, q_4\}, b) = [\delta(q_2, b)]_A = [q_3]_A = \{q_1, q_3\}$

Graphisch:



Aufgabe 5

Gesucht ist ein kontextfreie Grammatik G, die die Sprache

$$L \;=\; \{a^m b^n \mid m \neq n\}$$

akzeptiert. Die Idee besteht darin, zunächst gleich viele as und bs auf beiden Seiten des Startzeichens zu erzeugen und anschließend entweder weitere as oder weitere bs hinzuzufügen. Dazu wähle man $G = (\Sigma, N, S, P)$ mit $\Sigma = \{a, b\}$, $N = \{S, A, B\}$ und

$$\begin{array}{rrcll} P = \{ & S & \to & aSb & (1) \\ & S & \to & A & (2a) \\ & A & \to & aA & (3a) \\ & A & \to & a & (4a) \\ & S & \to & B & (2b) \\ & B & \to & Bb & (3b) \\ & B & \to & b\ \} & (4b) \end{array}$$

Es bleibt zu zeigen, dass $L = L(G)$.

'$\subseteq$': Sei $w = a^m b^n \in L$. Im Falle $m > n$ gilt

$$\begin{array}{rlll} S & \Rightarrow^n & a^n S b^n & \text{mit (1)} \\ & \Rightarrow & a^n A b^n & \text{mit (2a)} \\ & \Rightarrow^{m-n-1} & a^{m-1} A b^n & \text{mit (3a)} \\ & \Rightarrow & a^m b^n & \text{mit (4a)} \end{array}$$

und im Falle $m < n$

$$\begin{array}{rlll} S & \Rightarrow^n & a^n S b^n & \text{mit (1)} \\ & \Rightarrow & a^n B b^n & \text{mit (2b)} \\ & \Rightarrow^{n-m-1} & a^m B b^{n-1} & \text{mit (3b)} \\ & \Rightarrow & a^m b^n & \text{mit (4b)} \end{array}$$

Es gilt also in beiden Fällen $S \Rightarrow^* w$, d.h. $w \in L(G)$.

'$\supseteq$': Sei $w \in L(G)$, d.h. $S \Rightarrow^* w$. Der letzte Schritt dieser Ableitung kann nur mit (4a) oder (4b) erfolgen. Im ersten Fall muss irgendwann vorher ein Schritt mit (2a) erfolgt sein, also ist die gesamte Ableitung von der Form

$$\begin{array}{rlll} S & \Rightarrow^n & a^n S b^n & \text{mit (1)} \\ & \Rightarrow & a^n A b^n & \text{mit (2a)} \\ & \Rightarrow^k & a^{n+k} A b^n & \text{mit (3a)} \\ & \Rightarrow & a^{n+k+1} b^n & \text{mit (4a)} \end{array}$$

und damit $w = a^{n+k+1} b^n \in L$ wegen $n + k + 1 > n$.

Im zweiten Fall muss irgendwann vorher ein Schritt mit (2b) erfolgt sein, also ist die gesamte Ableitung von der Form

$$\begin{array}{rlll} S & \Rightarrow^n & a^n S b^n & \text{mit (1)} \\ & \Rightarrow & a^n B b^n & \text{mit (2b)} \\ & \Rightarrow^k & a^n A b^{n+k} & \text{mit (3b)} \\ & \Rightarrow & a^n b^{n+k+1} & \text{mit (4b)} \end{array}$$

und damit $w = a^n b^{n+k+1} \in L$ wegen $n < n + k + 1$.

Aufgabe 6

Es ist zu zeigen, dass die Sprache

$$L = \{a^n b^{2^n} \mid n \in \mathbb{N}\}$$

nicht kontextfrei ist. Nach dem starken Pumping Lemma für kontextfreie Sprachen genügt es zu zeigen, dass für jedes $n \geq 1$ ein Wort $z \in L$ mit $|z| \geq n$ existiert so, dass für alle Zerlegungen $z = uvwxy$ mit $vx \neq \varepsilon$ und $|vwx| \leq n$ ein Pumpfaktor $i \in \mathbb{N}$ existiert mit $uv^iwx^iy \notin L$.

Sei also $n \geq 1$. Man wähle $z = a^n b^{2^n}$. Dann ist $z \in L$ und $|z| = n+2^n \geq n$. Sei $z = uvwxy$ mit $vx \neq \varepsilon$ und $|vwx| \leq n$.

Wenn vx nur as enthält, dann ist $\#_a(uv^2wx^2y) = \#_a(z) + \#_a(vx) > n + \#_a(vx) > n$ und $\#_b(uv^2wx^2y) = \#_b(z) = 2^n$, also $uv^2wx^2y \notin L$.

Wenn vx mindestens ein b enthält, dann gilt $2^n = \#_b(z) < \#_b(uv^2wx^2y) = \#_b(z)+\#_b(vx) \leq 2^n+|vx| \leq 2^n+n < 2^n+2^n = 2^{n+1}$. Also liegt $\#_b(uv^2wx^2y)$ echt zwischen 2^n und 2^{n+1} und ist damit selbst keine Zweierpotenz, d.h. $uv^2wx^2y \notin L$.

Aufgabe 7

Eine Sprache L ist deterministisch kontextfrei, wenn es einen deterministischen Kellerautomaten M gibt, der die Sprache $L\$$ erkennt.

Sei $L = \{a^m b^n c^{m+n} \mid m, n \in \mathbb{N}\}$. Ein deterministischer Kellerautomat, der $L\$$ erkennt, arbeitet wie folgt: In der ersten Phase verschiebt er as in den Keller, in der zweiten bs, und in der dritten hebt er cs gegen as oder bs im Keller auf. Zur Unterscheidung der drei Phasen verwendet man drei Zustände, etwa s, q_b und q_c. Dabei ist s der Startzustand, in dem man die as liest, und q_b bzw. q_c signalisieren, dass man schon mindestens ein b bzw. c gelesen hat.

Zu beachten ist, dass nicht für jedes Wort $a^m b^n c^{m+n} \in L$ alle drei Phasen durchlaufen werden: Im Falle $n = 0$ und $m > 0$ entfällt die zweite Phase, d.h. man trifft schon im Zustand s auf ein c und muss in den Zustand q_c wechseln. Im Falle $m = n = 0$ trifft man im Zustand s sofort auf den Endmarker \$ und wechselt in den Endzustand f.

Diese Idee kann man mit dem folgenden Kellerautomaten realisieren: $M = (\Sigma, \Gamma, Q, s, F, \Delta)$ mit $\Sigma = \{a, b, c, \$\}, \Gamma = \{a, b, c\}, Q = \{s, q_b, q_c, f\}$, $F = \{f\}$ und

$$\begin{array}{lll}
\Delta = \{ \; ((s,a,\varepsilon),(s,a)), & (1) & a\text{s verschieben} \\
((s,b,\varepsilon),(q_b,b)), & (2) & \text{erstes } b \text{ verschieben, falls } n > 0 \\
((s,c,a),(q_c,\varepsilon)), & (3) & \text{erstes } c \text{ gegen } a \text{ aufheben, falls } n = 0 \\
((s,\$,\varepsilon),(f,\varepsilon)), & (4) & \text{sofort zu } f \text{ wechseln, falls } m = n = 0 \\
((q_b,b,\varepsilon),(q_b,b)), & (5) & \text{übrige } b\text{s verschieben} \\
((q_b,c,b),(q_c,\varepsilon)), & (6) & \text{erstes } c \text{ gegen } b \text{ aufheben, falls } n > 0 \\
((q_c,c,b),(q_c,\varepsilon)), & (7) & \text{übrige } c\text{s gegen } b\text{s aufheben} \\
((q_c,c,a),(q_c,\varepsilon)), & (8) & \text{übrige } c\text{s gegen } a\text{s aufheben} \\
((q_c,\$,\varepsilon),(f,\varepsilon)) \; \} & (9) & \text{fertig}
\end{array}$$

M ist ein deterministischer Kellerautomat, da die Transitionen in Δ paarweise inkompatibel sind: Zwei Transitionen, die vom gleichen Zustand ausgehen, unterscheiden sich entweder im Eingabezeichen oder—im Falle der beiden Transitionen (7) und (8)—im Kellerzeichen.

Es bleibt also noch zu zeigen, dass $L\$ = L(M)$.

'$\subseteq$': Sei $w = a^m b^n c^{m+n} \in L$. Es ist zu zeigen, dass $w\$ \in L(M)$, d.h. dass

$$(s, w\$, \varepsilon) \ \vdash_M^* \ (f, \varepsilon, \varepsilon)$$

Im Falle $m = n = 0$ ist $w = \varepsilon$ und damit

$$(s, w\$, \varepsilon) = (s, \$, \varepsilon) \ \vdash_M \ (f, \varepsilon, \varepsilon) \quad \text{mit (4)}$$

Im Falle $m > 0$ und $n = 0$ ist $w = a^m c^m$ und damit

$$\begin{array}{llll} (s, w\$, \varepsilon) = (s, a^m c^m \$, \varepsilon) & \vdash_M^m & (s, c^m \$, a^m) & \text{mit (1)} \\ & \vdash_M & (q_c, c^{m-1}\$, a^{m-1}) & \text{mit (3)} \\ & \vdash_M^{m-1} & (q_c, \$, \varepsilon) & \text{mit (8)} \\ & \vdash_M & (f, \varepsilon, \varepsilon) & \text{mit (9)} \end{array}$$

Man beachte, dass hier die Schreibweisen a^{m-1}, c^{m-1} und $\vdash_M^{m-1}$ nur deshalb erlaubt sind, weil $m > 0$ vorausgesetzt wurde.

Es bleibt der Fall $n > 0$ zu betrachten. In diesem Fall erhält man

$$\begin{array}{llll} (s, w\$, \varepsilon) = (s, a^m b^n c^{m+n} \$, \varepsilon) & \vdash_M^m & (s, b^n c^{m+n} \$, a^m) & \text{mit (1)} \\ & \vdash_M & (q_b, b^{n-1} c^{m+n} \$, b a^m) & \text{mit (2)} \\ & \vdash_M^{n-1} & (q_b, c^{m+n} \$, b^n a^m) & \text{mit (5)} \\ & \vdash_M & (q_c, c^{m+n-1} \$, b^{n-1} a^m) & \text{mit (6)} \\ & \vdash_M^{n-1} & (q_c, c^m \$, a^m) & \text{mit (7)} \\ & \vdash_M^m & (q_c, \$, \varepsilon) & \text{mit (8)} \\ & \vdash_M & (f, \varepsilon, \varepsilon) & \text{mit (9)} \end{array}$$

Hier ist zu beachten, dass die Schreibweisen b^{n-1}, c^{m+n-1} und $\vdash_M^{n-1}$ erlaubt sind, weil $n > 0$ vorausgesetzt wurde. Der Sonderfall $m = 0$ stört dagegen *nicht*, denn er bedeutet nur, dass die beiden Übergangsschrittfolgen mit (1) und (8) leer sind.

'$\supseteq$': Sei $z \in L(M)$, d.h.

$$(s, z, \varepsilon) \ \vdash_M^* \ (f, \varepsilon, \varepsilon) \qquad (*)$$

Wegen $s \neq f$ ist (*) eine nichtleere Folge von Übergangsschritten, deren letzter Schritt offensichtlich nur mit (4) oder (9) erfolgen kann.

Wenn der letzte Schritt mit (4) erfolgt, dann kann (*) nur von der Form

$$\begin{array}{llll} (s, z, \varepsilon) & \vdash_M^m & (s, \$, \varepsilon) & \text{mit (1)} \\ & \vdash_M & (f, \varepsilon, \varepsilon) & \text{mit (4)} \end{array}$$

sein. Da mit (1) jeweils ein a von der Eingabe in den Keller verschoben wird, ist dies nur möglich, wenn $m = 0$ und $z = \$$ ist, also $z = \varepsilon\$ \in L\$$.

Wenn der letzte Schritt von (*) mit (9) erfolgt, dann ist er von der Form

$$(q_c, \$, \varepsilon) \quad \vdash_M \quad (f, \varepsilon, \varepsilon)$$

also muss man sich überlegen, wie man von s nach q_c gelangen kann. Dafür gibt es zwei mögliche Wege. Entweder man gelangt mit (3) direkt von s nach q_c oder zuerst mit (2) von s nach q_b und von dort mit (6) nach q_c.

Im ersten Fall ist (*) von der Form

$$\begin{array}{lll l} (s, z, \varepsilon) & \vdash_M^m & (s, cu, a^m) & \text{mit } (1) \\ & \vdash_M & (q_c, u, a^{m-1}) & \text{mit } (3) \\ & \vdash_M^{m-1} & (q_c, \$, \varepsilon) & \text{mit } (7) \\ & \vdash_M & (f, \varepsilon, \varepsilon) & \text{mit } (9) \end{array}$$

Das ist nur möglich, wenn $m > 0$, $z = a^m cu$ und $u = c^{m-1}\$$ gilt und damit $z = a^m c^m \$ \in L\$$.

Im zweiten Fall ist (*) von der Form

$$\begin{array}{lll l} (s, z, \varepsilon) & \vdash_M^m & (s, bu, a^m) & \text{mit } (1) \\ & \vdash_M & (q_b, u, ba^m) & \text{mit } (2) \\ & \vdash_M^n & (q_b, cv, b^{n+1}a^m) & \text{mit } (5) \\ & \vdash_M & (q_c, v, b^n a^m) & \text{mit } (6) \\ & \vdash_M^{m+n} & (q_c, \$, \varepsilon) & \text{mit } (7) \text{ und } (8) \\ & \vdash_M & (f, \varepsilon, \varepsilon) & \text{mit } (9) \end{array}$$

Das ist nur möglich, wenn $z = a^m bu$, $u = b^n cv$ und $v = c^{m+n}\$$ gilt und damit $z = a^m b^{n+1} c^{m+n+1} \$ \in L\$$.

Insgesamt ist damit bewiesen, dass (*) nur für Wörter $z \in L\$$ gelten kann. Also ist $L(M) \subseteq L\$$.

Übrigens kann man in M eine Transition einsparen, indem man z.B. die as beim Verschieben in den Keller in bs umwandelt, d.h. indem man Transition (1) durch

$$((s, a, \varepsilon), (s, b))$$

ersetzt. Dann braucht man in der dritten Phase nur noch cs in der Eingabe gegen bs im Keller aufzuheben, d.h. Transition (3) ersetzt man durch

$$((s, c, b), (q_c, \varepsilon))$$

und Transition (8) entfällt.

Aufgabe 8

Siehe Vorlesungsmitschrift bzw. Folien.

Aufgabe 9

Gesucht ist ein `WHILE`-Programm, das die charakteristische Funktion der ungeraden natürlichen Zahlen berechnet.

Die Idee ist, dass wir wiederholt 1 von der Eingabe abziehen und dabei jeweils eine Hilfsvariable zwischen 0 und 1 wechseln lassen.

$$
\begin{array}{l}
\texttt{LOOP}\ x_1\ \texttt{DO} \\
\qquad x_3 \coloneqq\ 1; \\
\qquad \texttt{LOOP}\ x_2\ \texttt{DO}\ x_3 \coloneqq x_3 - 1\ \texttt{END}; \\
\qquad x_2 \coloneqq\ x_3; \\
\texttt{END} \\
x_1 \coloneqq\ x_2
\end{array}
$$

Aufgabe 10

Es ist zu zeigen, dass es keine totale universelle Funktion gibt; d.h. keine totale μ-rekursive Funktion $v : \mathbb{N}^2 \to \mathbb{N}$, so dass für jede totale μ-rekursive Funktion $h : \mathbb{N}^k \to \mathbb{N}$ ein Index e existiert mit

$$v(e, \langle x_1, \ldots, x_k \rangle) = h(x_1, \ldots, x_k)$$

Angenommen eine solche Funktion existiert. Dann ist die Funktion $g : \mathbb{N} \to \mathbb{N}$ definiert durch

$$g(n) = v(n, \langle n \rangle) + 1$$

ebenfalls total und μ-rekursiv. Also gibt es einen Index e, so dass

$$v(e, \langle n \rangle) = g(n)$$

Insbesondere gilt dann aber, wenn wir e für n einsetzen

$$v(e, \langle e \rangle) = g(e) = v(e, \langle e \rangle) + 1$$

und wir haben unseren gesuchten Widerspruch.

Aufgabe 11

Gesucht ist eine Funktion $g : \mathbb{N}^2 \to \mathbb{N}$, so dass

$$u(g(i,j), \cdot) = u(i, \cdot) + u(j, \cdot)\ .$$

Sei $h(i,j,n) = u(i,n) + u(j,n)$. Da h μ-rekursiv ist, gibt es einen Index e, so dass $u(e, \langle i,j,n \rangle) = h(i,j,n)$. Mit dem s-m-n-Theorem gilt dann:

$$u(i,n) + u(j,n) = h(i,j,n) = u(e, \langle i,j,n \rangle) = u(s(e, 2, \langle i,j \rangle), n)\ ;$$

also ist $g = \lambda i, j.s(e, 2, \langle i,j \rangle)$ die gesuchte Funktion.