Blatt 9

Aufgabe 1:

Es werden akzeptiert: $g^n s v^m h^k$ mit $n \leq m$ und $n \geq k$

Am Anfang stehen wir links des Eingabewortes.

Strategie: Lösche immer 1 Geschenk
1 Rentier
1 Haus

Am Ende dürfen dann nur noch Rentiere und Geschenke und Schlitten übrig sein.

Problem: Löschen heisst ein blank an die Stelle des Symbols schreiben.
Über blanks kann der Lesekopf aber nicht gehen ⇒ Anstatt ein Rentier zu löschen machen wir v' daraus...

$z_0 g \to z_1 \square R$ „Geschenk löschen, nach z_1, 1 nach rechts"
$z_1 g \to z_1 g R$ „alle restlichen Geschenke überspringen"
$z_1 s \to z_1 s R$ „den Schlitten überspr."
$z_1 v \to z_2 v' R$ „ein Rentier zu v'"
$z_1 v' \to z_1 v' R$ „Falls da schon v' ⇒ überspringen"
$z_2 v \to z_2 v R$ „restliche Rentiere überspr."
$z_2 h \to z_2 h R$ „alle Häuser überspringen"
$z_2 \square \to z_3 \square L$ „beim Ende eins zurück"
$z_3 h \to z_4 \square L$ „Letztes Haus löschen"

$z_4 h \to z_4 h L$
$z_4 v \to z_4 v L$
$z_4 v' \to z_4 v' L$ „alles überspringen und nach Links"
$z_4 s \to z_4 s L$
$z_4 g \to z_4 g L$
$z_4 \square \to z_0 \square R$ „von vorne los"

Sonderfälle bzw. Enden:

$z_0 S \to z_5 \square R$ „keine Geschenke mehr da ⇒ in z_5 dürfen nur noch Rentiere kommen!"

$z_5 v' \to z_5 \square R$

$z_5 v \to z_5 \square R$ „kommen da noch Häuser hängt er..."

$z_5 \square \to z_{Ende}$

$z_3\ v \to z_4 v L$ „keine Häuser mehr da ⇒ macht nix, es müssen nur noch Rentiere und Geschenke da sein

$z_3 v' \to z_4 v' L$

Fertig, mehr sollte nicht passieren können, wenn er akzeptieren soll...

Aufgabe 2:

$z_0 1000 \vdash 1 z_0 000 \vdash 10 z_0 00 \vdash^* 1000 z_0 \vdash 100 z_1 0$
$\vdash 10 z_1 01 \vdash 1 z_1 011 \vdash z_1 1111 \vdash z_2 \square 0111 \vdash z_e 0111$

Die Turingmaschine subtrahiert von der geg. Binärzahl 1.

Aufgabe 3:

a) Der nichtdeterministische LBA hat den Anfangszustand z, Endzustand z_e, Bandalphabet $\{0, 1, 2, \#, \$, ¢\}$

z	$\$$	$\to$	z	$\$$	R	
z	i	$\to$	z	i	R	$i \in \{0,1\}$
z	i	$\to$	z_i	2	L	Anfang der 2. Hälfte raten
z_i	j	$\to$	z_i	j	L	$j \in \{0,1,2\}$ nach links gehen bis
z_i	k	$\to$	z_i'	k	R	$k \in \{\#, \$\}$ zum ersten Sonderzeichen
z_i'	i	$\to$	z_2	$\#$	R	Vergleich
z_2	i	$\to$	z_2	i	R	Nach rechts
z_2	2	$\to$	z_3	2	R	bis 2,
z_3	2	$\to$	z_3	2	R	2-er überlaufen,
z_3	i	$\to$	z_i	2	R	nächstes Zeichen löschen und merken
z_3	$¢$	$\to$	z_4	$¢$	L	Ende des 2. Wortes
z_4	2	$\to$	z_4	2	L	2-er überlaufen,
z_4	$\#$	$\to$	z_e	$\#$	L	auch 1. Wort ist zu Ende.

Die Grammatik hat die Produktionen

$S \to 00 \mid 11 \mid 0NS \mid 1ES \mid A0 \mid B1$
$E0 \to 0E,\ E1 \to 1E,\ N0 \to 0N,\ N1 \to 1N$
$EA \to A1,\ EB \to B1,\ NA \to A0,\ NB \to B0$
$A \to 0,\ B \to 1$

a) Der nichtdeterministische LBA hat den Anfangszustand z, Endzustand z_e, Bandalphabet $\{0,1,2,\#,\$,¢\}$

z	$\$$	$\rightarrow$	z	$\$$	R	
z	i	$\rightarrow$	z	i	R	$i \in \{0,1\}$
z	i	$\rightarrow$	z_i	2	L	Anfang der 2. Hälfte raten
z_i	j	$\rightarrow$	z_i	j	L	$j \in \{0,1,2\}$ nach links gehen bis
z_i	k	$\rightarrow$	z_i'	k	R	$k \in \{\#,\$\}$ zum ersten Sonderzeichen
z_i'	i	$\rightarrow$	z_2	$\#$	R	Vergleich
z_2	i	$\rightarrow$	z_2	i	R	Nach rechts
z_2	2	$\rightarrow$	z_3	2	R	bis 2,
z_3	2	$\rightarrow$	z_3	2	R	2-er überlaufen,
z_3	i	$\rightarrow$	z_i	2	R	nächstes Zeichen löschen und merken
z_3	¢	$\rightarrow$	z_4	¢	L	Ende des 2. Wortes
z_4	2	$\rightarrow$	z_4	2	L	2-er überlaufen,
z_4	$\#$	$\rightarrow$	z_e	$\#$	L	auch 1. Wort ist zu Ende.

Die Grammatik hat die Produktionen

$$S \rightarrow 00 \mid 11 \mid 0NS \mid 1ES \mid A0 \mid B1$$
$$E0 \rightarrow 0E,\ E1 \rightarrow 1E,\ N0 \rightarrow 0N,\ N1 \rightarrow 1N$$
$$EA \rightarrow A1,\ EB \rightarrow B1,\ NA \rightarrow A0,\ NB \rightarrow B0$$
$$A \rightarrow 0,\ B \rightarrow 1$$

b) $S \to SA \mid 1f1$
$1A \to A11$
$fA \to 0f \mid 1f1$

Idee: Wandere nach rechts, lösche da eine 1 und lauf anschließend zum f und wende auf den Teil links davon den binären Subtraktionsalgorithmus aus A2 an.

$z_0 0 \to z_0 0 R$
$z_0 1 \to z_0 1 R$
$z_0 f \to z_0 f R$ „nach rechts"
$z_0 \square \to z_1 \square L$
$z_1 1 \to z_2 \square L$ „eine 1 löschen"
$z_2 1 \to z_2 1 L$
$z_2 f \to z_3 f L$
$z_3 0 \to z_3 1 L$ „der Algorithmus von Aufgabe 2"
$z_3 1 \to z_4 0 L$
$z_4 0 \to z_4 0 L$
$z_4 1 \to z_4 1 L$
$z_4 \square \to z_0 \square R$

Ende:
$z_1 f \to z_5 f L$ „Jetzt sind keine 1 mehr rechts des f
$\Rightarrow$ links davon dürfen nur noch 0 sein"

$z_5 0 \to z_5 0 L$
$z_5 \square \to z_6 \square R$
z_6 = akz. Zust.

d) $S \Rightarrow A$
$S \Rightarrow SB$
$A \Rightarrow a$
$AB \Rightarrow Aa$
$aB \Rightarrow Baa$

Idee: 1^{2^n} heisst verdoppeln und zwar n mal
$\Rightarrow$ wir halbieren und wenn am Ende 1 übrigbleibt war das Wort o.k.

$z_0 a \rightarrow z_1 a'$ „erstes a markieren"
$z_1 a \rightarrow z_1 a R$
$z_1 \Box \rightarrow z_2 \Box L$
$z_2 a \rightarrow z_3 \Box L$ „letztes a löschen"
$z_3 a \rightarrow z_3 a L$
$z_3 a' \rightarrow z_4 a' R$
$z_4 a \rightarrow z_1 a' R$ „nächstes a markieren und von vorne"
$z_4 \Box \rightarrow z_5 \Box L$ „wir haben halbiert"
$z_5 a' \rightarrow z_5 a L$ „alle „unmarkieren" "
$z_5 \Box \rightarrow z_0 \Box R$ „wieder neu starten"

Bleibt noch das Ende: nur noch 1 a da.
$z_2 a' \rightarrow z_{Ende}$

d) Wie kann man die Quadratzahlen rekursiv erzeugen?

$(n+1)^2 = n^2 + 2n + 1$ also: $2\times$ Anzahl Quadratzahlen in n^2 $+1$

grafisch:

1^2 $\quad 2^2$ $\quad 3^2$ $\quad 4^2$

$\quad 1+3$ $\quad 4+5$ $\quad 9+7$

Wenn man also immer das eine zusätzliche markiert, weiss man, wieviele Quadrate in n^2 stecken.

Vorgehen daher:

a a a a a a a a ...

↑
markieren

â a a ...

Jetzt rekursiv von vorne durchgehen und pro markiertes â hinter dem letzten â zwei a's zu â machen, bis man das für alle â gemacht hat (=> muss sich merken, welche â schon hinten angehängt wurden).
Anschließend das nächste a nach den â zu â machen.
Die ã wieder zu a und auch die markierten â wieder zurücksetzen.

Ist am Ende das letzte a markiert (â), war die Eingabe o.k.

Bsp.

$\hat{a}\ a\ a\ a\ a\ a\ a\ a\ a$

$\bar{\hat{a}}\ \tilde{a}\ \tilde{a}\ a\ \ a\ a\ a\ a\ a$

$\bar{\hat{a}}\ \tilde{a}\ \tilde{a}\ \hat{a}\ a\ a\ a\ a\ a$

$\hat{a}\ a\ \ a\ \hat{a}\ a\ a\ a\ a\ a$

$\bar{\hat{a}}\ a\ \ a\ \bar{\hat{a}}\ \tilde{\hat{a}}\ \tilde{a}\ \tilde{a}\ \tilde{a}\ a$

$\bar{\hat{a}}\ a\ \ a\ \bar{\hat{a}}\ \hat{a}\ \tilde{a}\ \tilde{a}\ \tilde{a}\ \hat{a}$ fertig

Grammatik:

$S \Rightarrow ABS \mid C$

$B \Rightarrow a,\ C \Rightarrow a,\ D \Rightarrow a$

$AB \Rightarrow DDBA$

$AD \Rightarrow DA$

$ABC \Rightarrow DDBC$

$(n+1)^2 = n^2 + 2n + 1$

3.4

q_0	1	$\rightarrow$		*stop*		Binärwort fertig
q_0	b	$\rightarrow$	q_0	b	R	
q_0	0	$\rightarrow$	q_1	b	R	Eine 0 löschen
q_1	0/1	$\rightarrow$	q_1	0/1	R	Rechtes Ende aufsuchen
q_1	b	$\rightarrow$	q_2	b	L	
q_2	0	$\rightarrow$	q_3	1	L	Binäraddition um 1
q_2	1	$\rightarrow$	q_2	0	L	
q_2	b	$\rightarrow$	q_0	1	L	Fertig, da keine 0 mehr
q_3	0	$\rightarrow$	q_3	0	L	Linkes Ende aufsuchen
q_3	b	$\rightarrow$	q_1	b	R	keine 0 mehr löschen, da führende 0 ersetzt
q_3	1	$\rightarrow$	q_4	1	L	eine 0 ist zu löschen
q_4	0/1	$\rightarrow$	q_4	0/1	L	Linkes Ende weitersuchen
q_4	b	$\rightarrow$	q_0	b	R	Nächster Durchgang

Aufgabe 5:

a) $L = \{a^i b^j c^i \mid i,j \geq 0\} \cup \{a^i b^j d^j \mid i,j \geq 0\}$

z.z.: L ist deterministisch kontextfrei

⇒ determ. Kellerautomat

⇒ $|\delta(z,a,A)| + |\delta(z,\varepsilon,A)| \leq 1$

d.h. im Zustand z mit oberstem Kellerzeichen A kann entweder ein ε-Übergang oder ein Übergang mit einem Eingabezeichen a erfolgen.

$z_0 a \#$	$\to z_0 A\#$	„a's kellern"
$z_0 a A$	$\to z_0 AA$	
$z_0 b A$	$\to z_1 BA$	„nach den a's kommen b's ⇒ kellern"
$z_1 b B$	$\to z_1 BB$	
$z_1 c B$	$\to z_2 \varepsilon$	„nach den b's kommen c's" c
$z_2 \varepsilon B$	$\to z_2 \varepsilon$	„⇒ B's aus dem Keller raus"
$z_2 \varepsilon A$	$\to z_3 \varepsilon$	„das A für das gelesene erste c raus"
$z_3 c A$	$\to z_3 \varepsilon$	
$z_3 \varepsilon \#$	$\to z_{Ende} \varepsilon$	„die A's und c's gingen auf"
$z_1 d B$	$\to z_4 \varepsilon$	„nach den b's kommen d's" d
$z_4 d B$	$\to z_4 \varepsilon$	
$z_4 \varepsilon A$	$\to z_{Ende} \varepsilon$	„es gab a's und die d's gingen mit B's auf"
$z_4 \varepsilon \#$	$\to z_{Ende} \varepsilon$	„es gab keinerlei a's"

Wie eben gesehen, können also auch 0 a's vorkommen:

$z_0 b \#$	$\to z_5 B\#$	„es gab keine a's"
$z_5 b B$	$\to z_5 BB$	
$z_5 d B$	$\to z_4 \varepsilon$	

Ebenso können auch 0 b's vorkommen:

$z_0 c A \to z_3 \varepsilon$

Endzustände sind:

z_{Ende}	
z_0	„$a^0 b^0 c^0$, $a^* b^0 d^0$"
z_5	„$a^0 b^* c^0$"

$M = (\{z_0, z_1, \ldots, z_5, z_{Ende}\}, \{a,b,c,d\}, \{A,B,\#\}, \text{s.o.}, z_0, \#, \{z_{Ende}, z_0, z_5\})$

$= (Z, \Sigma, \Gamma, \delta, z_s, \#, E)$

b). Widerspruchsbeweis: Annahme: „Es gibt so einen Automaten"

⇒ ↯ Muss Fehler machen, also falsche Wörter erkennen oder richtige verwerfen.

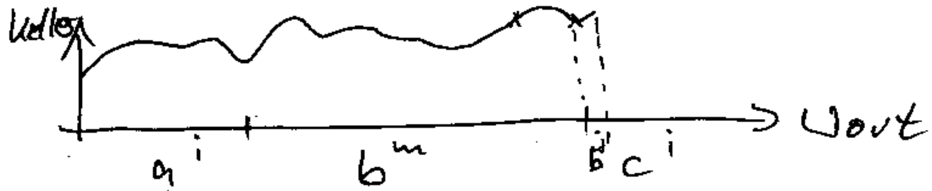
$\exists\, k \neq l, j'$

$a^k b^{j(k)+j'} c^k$

$a^l b^{j(l)+j'} c^l$

Der Automat ist deterministisch ⇒ Es gibt nur endlich viele Tupel aus Zustand × oberstes Kellerz

⇒ Es gibt mindestens zwei Vorkommen einer Konfig.



⇒ Wenn ich meine Fkt. $j(x)$ entsprechend wähle, lande ich beide Male ($a^k b^{j(k)+j'} c^k$ / $a^l b^{j(l)+j'} c^l$) vor dem c-Block in der gleichen Konfiguration

⇒ ↯ denn hänge ich an $a^k b^{j(k)+j'} c^l$ lande ich in derselben Konfiguration, wie bei $a^l b^{j(k)+j'} c^l$ und nur das darf akzeptiert werden... (Eine einzige Konfig. kann nicht falsch und richtig sein!)

Anschließend Muster ⇒

b) Ein deterministischer Kellerautomat ist nach Lesen von a^i in der Konfiguration $\delta(q_0, a^i) = (q(i), x_{i,1} \ldots x_{i,n(i)})$. Bezüglich diesem i ist nun der maximale Abbau des Kellerinhaltes bei nachfolgenden $b's$ wie folgt definiert:

$$m(i) := \max_j \{m \mid \delta(q_0, a^i b^j) = (p(i), x_{i,m} \ldots x_{i,n(i)})\}$$

Sei $j(i) := \min\{j \mid \delta(q_0, a^i b^j) = (p(i), x_{i,m(i)} \ldots x_{i,n(i)})\}$ und $f(i) := (p(i), x_{i,m(i)})$ für $\delta(q_0, a^i b^{j(i)}) = (p(i), x_{i,m(i)} \ldots x_{i,n(i)})$. Da es nur endlich viele Zustände und Kellerzeichen gibt, gibt es $k \neq l$ mit $f(k) = f(l)$. Damit beliebig viele d^j unterschieden werden können, gibt es unendlich viele obere Kellerinhalte $\gamma(j')$ mit

$$\begin{aligned} \delta(q_0, a^k b^{j(k)+j'}) &= (p, \gamma(j') x_{k,m(k)} \ldots x_{k,n(k)}) \text{ und} \\ \delta(q_0, a^l b^{j(l)+j'}) &= (p, \gamma(j') x_{l,m(l)} \ldots x_{l,n(l)}). \end{aligned}$$

Daher gibt es ein j' mit $|\gamma(j')| > max(k, l)$ und somit kann c^k und c^l nicht mehr unterschieden werden, da $\gamma(j')$ beim Lesen von c^k oder c^l nicht vollständig ausgekellert werden kann.