Einführung in die Theoretische Informatik B
Lösungsvorschläge zur
Klausur am 16. September 2006
Sommersemester 2006

Aufgabe 1

(i) Welche der folgenden Aussagen ist/sind korrekt?

korrekt	falsch	
[X]	[ ]	Sei M eine beliebige determinierte TM über dem Ein-/Ausgabealphabet Σ. Dann gilt $L_M = \{x \in \Sigma^* \mid f_M(x) = \varepsilon\}$.
[X]	[ ]	Sei M eine beliebige determinierte TM. Dann gilt Def($\tilde{s}_M$)=Def($\tilde{t}_M$).
[ ]	[X]	Aus $f \in O(n)$ folgt $f \in o(n)$.
[ ]	[X]	Die Funktion $f(n) = \lceil\sqrt{n}\rceil$ ist zeitkonstruierbar.
[X]	[ ]	Sei M eine Kontrollturingmaschine mit Eingabealphabet Σ. Wenn es für $x \in \Sigma^*$ ein $y \in \{0,1\}^*$ gibt, so dass $\tau_M(x,y) = 1$ gilt, dann gilt $x \in L_M$.

(ii) Welche der folgenden Aussagen ist/sind korrekt?

korrekt	falsch	
[X]	[ ]	Für alle $A \in$ NBAND(n) gilt $A \in$ BAND(n^2).
[ ]	[X]	Es ist unbekannt, ob P $\subseteq$ NP gilt.
[X]	[ ]	3SAT$\leq_{pol}$SAT
[ ]	[X]	PSPACE ist nicht abgeschlossen bzgl. $\leq_{pol}$.
[X]	[ ]	Aus $A \leq_{log} B$ und $B \in$NP folgt $A \in$NPSPACE.
[X]	[ ]	Es ist bekannt, dass NLOGSPACE$\subseteq$P gilt.
[X]	[ ]	P$\neq$NP$\Leftrightarrow$ 3SAT $\notin$ P
[ ]	[X]	Es ist bekannt, dass P=LOGSPACE gilt.

(iii) Welche der folgenden Aussagen ist/sind korrekt?

korrekt	falsch	
[X]	[ ]	Die Sprache $\{a^n b^n \mid n \in \mathbb{N}\}$ ist kontextfrei.
		G ist eine rechtslineare Grammatik.
[X]	[ ]	Sei $G = (\{S,T\},\{a,b\},\{S \to aT\|T\,,\ T \to aT\|S\},S)$.
		G ist eine kontextfreie Grammatik.
[ ]	[X]	Sei $G = (\{S,T\},\{a,b\},\{S \to aSb\|T\,,\ T \to aT\|\varepsilon\},S)$.
		Es ist L(G)=$\{a^m b^n \mid m,n \in \mathbb{N}, m > n\}$.
[ ]	[X]	Sei $G = (\{S,T\},\{a,b\},\{S \to aaSb\|T\,,\ T \to aTb\|ab\},S)$.
		Es ist L(G)=$\{a^m b^n \mid m,n \in \mathbb{N}\backslash\{0\}, m \geq n\}$.

Aufgabe 2

Die Sätze können in den Kursunterlagen nachgelesen werden.

Aufgabe 3

Die Definitionen können in den Kursunterlagen nachgelesen werden.

Aufgabe 4

(i) Wir beschreiben die Arbeitsweise einer TM M mit $f_M(0^n) = 0^{f(n)}$ und $\tilde{s}_m \in O(f)$. M arbeitet auf der Eingabe $w := 0^n$ wie folgt:

1.) Teste, ob die Länge von w gerade ist. Falls ja, dann gebe w aus und halte. Falls nein, gehe zu 2.)

2.) Ermittle die Länge der Eingabe und stelle sie in dualer Form da (siehe Beispiel 2.1.2 aus KE 1). Berechne also $w' := d(lg(w))$. Gebe $0^{lg(w')}$ aus.

Es ist

$$f_M(w) = \begin{cases} w & , \text{ falls } lg(w) \text{ gerade} \\ 0^{lg(d(lg(w)))} & , \text{ sonst} \end{cases}$$

Da $lg(d(lg(0^n))) = \log\ n$ gilt, folgt $f_M = f$.
Bandbedarf bei Eingabe 0^n:

1.) Es ist kein zusätzliches Band nötig. Also $O(1)$.

2.) $O(\log\ n)$

somit $\tilde{s}_M \in O(log\ n) \subseteq O(f)$.

(ii) $g : \mathbb{N} \to \mathbb{N}$

$$n \mapsto \begin{cases} n & , \text{ falls } \varphi_n(n) \text{ def.} \\ n+1 & , \text{ sonst.} \end{cases}$$

Offensichtlich gilt $n \in O(g)$. Wäre g zeitkonstruierbar, dann wäre g berechenbar und K_φ entscheidbar. Widerspruch!

Aufgabe 5

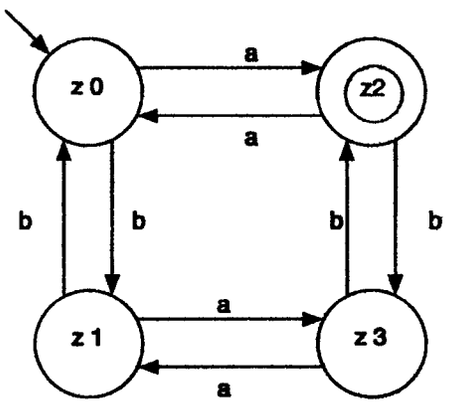
(i) f ist bandkonstruierbar (siehe Aufgabe 4 (i)). Es gilt $f \notin O(\log\ n)$. Mit dem Bandseparationssatz folgt BAND $(f) \not\subseteq$ BAND $(\log\ n)$. (Bem.: (i) folgt auch direkt aus (ii)).

(ii) f ist bandkonstruierbar, $\log \in O(f), f \notin O(\log\ n)$. Mit dem Bandhierarchiesatz folgt die zu beweisende Aussage.

Aufgabe 6

(i) Wir definieren

$M = (\{z_0, z_1, z_2, z_3\}, \{a, b\}, \delta, z_0, \{z_2\})$ mit der Überführungsfunktion definiert durch den folgenden Übergangsgraphen



Die Ideen, die mit den Zuständen verbunden sind, können, wie folgt zusammengefasst werden

z_0 Die Anzahl der bisher eingelesenen a's und b's ist gerade.
z_1 Die Anzahl der bisher eingelesenen a's ist ungerade, die der b's dagegen gerade.
z_2 Die Anzahl der bisher eingelesenen a's ist gerade, die der b's dagegen ungerade.
z_3 Die Anzahl der bisher eingelesenen a's und b's ist ungerade.

$S \to aA \mid bB, A \to aS \mid bC \mid \varepsilon, B \to bS \mid aC,$
$C \to aB \mid bA.$
G hat 9 Regeln.

(iii) $\alpha = (b^*ab^*ab^*)^*$

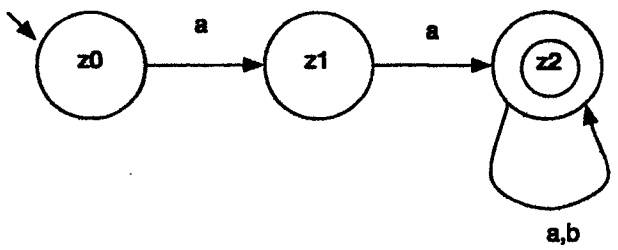
Aufgabe 7

(i) $L(G) = L(\alpha)$ mit $\alpha = aa(a \cup b)^*$.

(ii) Wir geben einen Automaten A an mit $L(A) = L(G)$ und benutzen dabei das im Kurs vorgestellte Verfahren:

Wir gehen in drei Schritten vor.

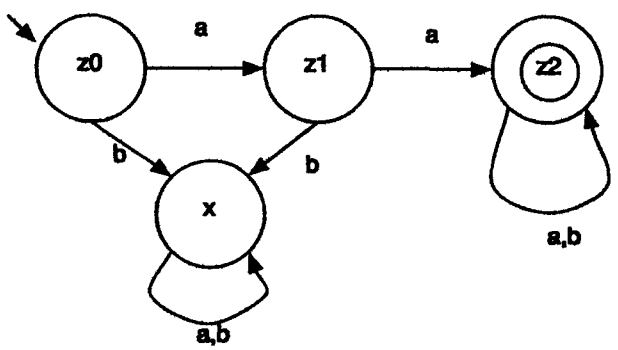
Schritt 1 Wir konstruieren einen determinierten endlichen Automaten A, so dass $T(A) = L(G)$ für die Grammatik G ist. Der Übergangsgraph δ sieht wie folgt aus



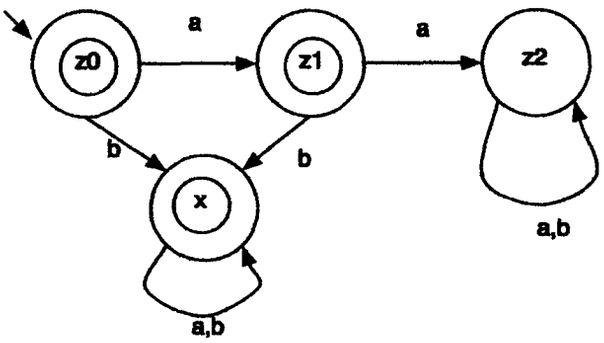
Schritt 2 Entsprechend von Satz 8.4.1 wissen wir, dass reguläre Sprachen unter Komplementbildung abgeschlossen sind.

A ist nicht vollständig, wir vervollständigen ihn. Sei A' der entstandene Automat:

Für A' gilt $L(A) = L(A')$.



Nun vertauschen wir Endstände und Nicht-Endstände



Sei A'' der resultierende Automat. Dann gilt $L(A'') = \{a,b\}^* \backslash L(A) = \{a,b\}^* \backslash L(G)$.

Schritt 3 Die dazugehörige Regelmenge kann nun aus dem Übergangsgraphen abgelesen werden

$$
\begin{array}{lcll}
R' & : & A & \rightarrow & \epsilon \\
 & & A & \rightarrow & a \\
 & & A & \rightarrow & b \\
 & & A & \rightarrow & aB \\
 & & A & \rightarrow & bC \\
 & & B & \rightarrow & b \\
 & & B & \rightarrow & bC \\
 & & C & \rightarrow & a \\
 & & C & \rightarrow & b \\
 & & C & \rightarrow & aC \\
 & & C & \rightarrow & bC
\end{array}
$$

und damit ist $G' = (\{A,B,C\}, \{a,b\}, R', A)$.

Aufgabe 8

(i) Die Idee für die Konstruktion des Kellerautomaten ist, zunächst die Mitte des Wortes zu (zufällig) zu raten und dabei die Anzahl der Zeichen bis zu dieser geratenen Mitte mit Hilfe des Kellers zu zählen.

Anschließend wird geprüft, ob das Restwort die gleiche Länge besitzt und mindestens zwei Symbole b enthält.

Damit ergibt sich der folgende Automat:

$M = (\{a,b\}, \{1,\#\}, \#, \{q_0, q_1, q_2, q_3\}, q_0, \{q_3\}, \delta)$ mit der folgenden Übergangsfunktion δ. Dazu setze $x \in \{a,b\}$

$(q_0, x, \#, q_0, 1)$	Beginn, um zu zählen
$(q_0, x, 1, q_0, 1)$	Zeichen zählen
$(q_0, \epsilon, 1, q_1, 1)$	Wortmitte raten
$(q_1, a, 1, q_1, \epsilon)$	Länge des Restwortes (noch kein b)
$(q_1, b, 1, q_2, \epsilon)$	Erstes b im Restwort
$(q_2, a, 1, q_2, \epsilon)$	Länge des Restwortes (bisher nur ein b)
$(q_2, b, 1, q_3, \epsilon)$	Zweites b im Restwort
$(q_3, x, 1, q_3, \epsilon)$	Länge des Restwortes
$(q_3, \epsilon, \#, q_3, \epsilon)$	Keller leeren

(ii) Es gibt mehr als eine nicht-akzeptierende Konfigurationsfolge für das Wort $ababab$ ($\in T(M)$) und zwar wenn die Wortmitte falsch geraten wird. Ein solche mögliche nicht-akzeptierende Konfigurationsfolge ist

$(q_0, ababab, \#) \vdash (q_0, babab, 1\#) \vdash (q_0, abab, 11\#) \vdash (q_1, abab, 11\#) \vdash (q_1, bab, 1\#) \vdash (q_2, b, \#)$

(iii) Die Sprache $L = \{xy \in \{a,b\}^* \mid lg(x) = lg(y) \wedge \sharp_b(y) >, 2\}$ ist <u>nicht</u> regulär (aber kontextfrei).
Das beweisen wir durch die Anwendung des Pumping Lemmas.
Angenommen L sei regulär.
Dann gibt es eine Zahl $n \in \mathbb{N}$, so dass alle $t, \bar{t}, z \in \Sigma^*$ mit $tz\bar{t} \in L$ und $lg(z) = n$ gilt $\exists n, v, w \in \Sigma^* z = uvw \ \wedge \ v \neq \epsilon \wedge \forall i \geq 0. tuv^iw\bar{t} \in L\}$

Wir betrachten die Worte <u>mit mindestens</u> zwei Zeichen b

(1) bb mit der Länge 2
(2) $bxby$ mit der Länge $2n + 2$
(3) $xybb$
(4) $xybb$
(5) $xbyb$

In der folgenden Betrachtung können wir die Fälle (2)-(5) zusammenfassend untersuchen.

<u>zu (1)</u> Die entsprechende Zerlegung $tZ\bar{t}$ mit $z = uvw, v \neq \epsilon$ ergibt, dass $v = bc$ mit $c \in \Sigma^*$. Dann wäre auch das Wort $uv = b^1c^{n-|v|}b^1$ in der Sprache, was mit $lg(bc)$ ungerade im Widerspruch zur Definition von L steht.

<u>zu (2)-(5)</u> Auch hier brauchen wir tatsächlich nur die Längen der Teilworte zu betrachten. Die entsprechende Zerlegung unter Anwendung des Pumping Lemmas, wie oben, zeigt, dass $lg(v) = n + 1$. Dann gälte auch für die Länge des (Teil-) Wortes uv $(2n + 2) - (n - 1) = n + 1$. Das ist ebenfalls ein Widerspruch zur Definition von L, wenn n gerade ist. Daher ist L <u>nicht</u> regulär.

BEOBACHTUNG:
Die Anzahl $\sharp_b \geq 2$ ist nicht wesentlich für diesen Beweis. Wichtig ist die Längenbedingung $lg(x) = lg(y)$, d.h. $lg(x) + lg(y)$ ist $2n$ für alle $n \in \mathbb{N}$.

Aufgabe 9

Sei $G = (U, E)$ nicht k-vernetzt. Dann gibt es $K \subseteq V$ mit $\sharp K = k$, die keine Kante in E enthält; K ist also eine unabhängige Menge mit k Knoten. Damit gilt für alle $x \in \{0,1\}^*$:

$$x\sharp^k \in \text{IND-SET} \Leftrightarrow x\sharp^k \notin M.$$

a) $M \notin$ P (weil $\overline{M}$ NP-vollständig und vermutlich P$\neq$NP)
$M \notin$ NP (weil NP vermutlich nicht unter Komplement abgeschlossen)
$M \in$ PSPACE (weil NP $\subseteq$PSPACE und PSPACE abgeschlossen ist unter Komplementbildung).

b) Sei $K, L \subseteq \Sigma^*$, $f : \Sigma^* \to \Sigma^*$. Dann gilt:

$$(\forall x. x \in K \Leftrightarrow f(x) \in L) \Leftrightarrow (\forall x.\ x \in \Sigma^* \backslash K \Leftrightarrow f(x) \in \Sigma^* \backslash L).$$

Daraus folgt:

$$K \leq_{pol} L \Leftrightarrow \Sigma^* \backslash K \leq_{pol} \Sigma^* \backslash L.$$

<u>Annahme:</u> M NP-vollständig.
Sei $K \in$ NP, $K \subseteq \Sigma^*, \Sigma \supseteq \{0, 1, \sharp\}$. Dann gilt
$K \leq_{pol} M$ (nach Annahme)
$\Rightarrow \Sigma^* \backslash K \leq_{pol} \Sigma^* \backslash M$ (s.o.)
$\leq_{pol}$IND-SET $= (\Sigma^* \backslash M) \cap \{0,1\}^* \{\sharp\}^*$
$\Rightarrow \Sigma^* \backslash K \in$ NP (da IND-SET NP-vollständig).