Kurs 01658 — Grundlagen der Theoretischen Informatik B

Lösungsvorschläge zur Klausur vom 28. 07. 2012

Aufgabe 1 (a) Die Aussage trifft zu. Satz 2.5.3 gilt nämlich entsprechend auch für FZEIT und FBAND. (Der Beweis ist derselbe wie für ZEIT und BAND, wie sofort aus der Definition 2.4.1 der Komplexitätsklassen folgt.)

(b) Die Aussage trifft *nicht* zu. Die erste Komplexitätsklasse umfasst *Funktionen,* die zweite *Sprachen.*

(c) Die Aussage trifft *nicht* zu. Die erste Komplexitätsklasse umfasst *Wortfunktionen,* die zweite *Funktionen von* $\mathbb{N}$ *nach* $\mathbb{N}$.

(d) Die Aussage trifft zu. Es handelt sich um einen Spezialfall der zweiten Aussage von Satz 2.5.3 (mit $f = \log$), der in Satz 3.5.2 noch einmal explizit formuliert worden ist.

Aufgabe 2 (a) (1) Wir beschreiben die Arbeitsweise einer geeigneten 2-Band-Turingmaschine M bei Eingabe von $\mathtt{0}^n$:

1. Berechne (durch duales Inkrementieren) $a = d(n)$ auf Band 2 und Band 3;
2. Überschreibe jedes Symbol $\neq \mathtt{B}$ auf Band 2 mit „0“;
3. Positioniere den Kopf von Band 3 auf das (von links) erste Symbol $\neq \mathtt{B}$;
4. WHILE *aktuelles Symbol auf Band* $3 \neq \mathtt{B}$ DO
   Kopiere den Inhalt von Band 2 auf das Ausgabeband;
   Gehe auf Band 3 einen Schritt nach rechts.

Die Korrektheit der Maschine braucht hier nicht bewiesen zu werden.

(2) Zur Abschätzung des Bandbedarfs geben wir für jedes Band ein entsprechendes „$O(\cdots)$“ an:

2: $O(\log n)$;
3: $O(\log n)$.

Man beachte, dass nur die Arbeitsbänder zur Bandkomplexität beitragen. Es folgt $\tilde{s}_M \in O(\log n)$. Also kommt die Maschine sogar mit weniger als dem zulässigen Speicherplatz aus. Damit ist f bandkonstruierbar.

(b) Angenommen, es gälte $(\log n)^2 \in O(\log n)$. Dann gibt es ein $c \in \mathbb{N}$ mit $\forall n \in \mathbb{N}\,.\,(\log n)^2 \leq c \cdot \log n + c$. Wir betrachten die Zahl $n_0 := 2^{2c+1}$. Damit gilt:

$$\begin{aligned} 2c+1 = \log_2(2^{2c+1}) \leq \log(2^{2c+1}) &= \frac{\left(\log(2^{2c+1})\right)^2}{\log(2^{2c+1})} \\ &\leq \frac{c \cdot \log(2^{2c+1}) + c}{\log(2^{2c+1})} \leq \frac{2c \cdot \log(2^{2c+1})}{\log(2^{2c+1})} = 2c\,, \end{aligned}$$

Dies ist ein Widerspruch. Es folgt $(\log n)^2 \notin O(\log n)$.

(c) Die Antwortet lautet „ja“. Zur Begründung definieren wir $f, g : \mathbb{N} \to \mathbb{N}$ durch $f(n) := \log(n)$ und $g(n) := \left(\log(n)\right)^2$ für alle $n \in \mathbb{N}$. Wegen der Gültigkeit von

$$\forall n \in \mathbb{N}\,.\,\log(n) \leq f(n) \leq g(n)$$

erhalten wir $\log \in O(g)$ und $f \in O(g)$. Nach (b) gilt $g \notin O(f)$.

Laut Aufgabenteil (a) ist die Funktion g bandkonstruierbar. Damit sind alle Voraussetzungen des Bandhierarchiesatzes 3.4.5 erfüllt. Es folgt $\mathsf{BAND}(f) \subsetneq \mathsf{BAND}(g)$. Somit existiert eine Sprache $L \in \mathsf{BAND}(g) \setminus \mathsf{BAND}(f)$.

Aufgabe 3 1. (a) Richtig. (Umformulierung der Definition.)

(b) Richtig. (Definition der NP-Vollständigkeit.)

(c) Richtig. (Lemma 4.3.3.)

(d) Falsch. (f muss lediglich total sein.)

(e) Richtig. (Es wird insbesondere über alle $\mathsf{NZEIT}\left(n^k\right)$ vereinigt, und andererseits ist jedes f, über das vereinigt wird, durch ein n^k größenordnungsmäßig beschränkt.)

(f) Falsch. (Die Menge ist offensichtlich sogar in P.)

2. (a) Richtig. (Die Maschinenbegriffe sind unterschiedlich.)

(b) Richtig. (Folgt aus Satz 4.2.1.)

(c) Richtig. (Folgt aus Satz 4.2.3 (1) und der Tatsache, dass jede rekursive Menge auch rekursiv-aufzählbar ist.)

(d) Richtig. (Folgt aus dem Bandhierarchiesatz und Satz 4.2.3 (2).)

(e) Falsch. (Nur *Sprachen* können aufeinander reduzierbar sein.)

(f) Falsch. (Satz 4.2.3 (2).)

Aufgabe 4 **(a)** Wir modifizieren die Kontrollturingmaschine T aus dem Kurstext, die die Menge RUCKSACK erkennt, geeignet:

Es sei x die Eingabe auf Band 1, und $y = y_1 y_2 \dots y_n \dots$, wobei $y_i \in \{0,1\}$ für alle $i \leq \lg(y)$ gelte, sei die Hilfseingabe auf Band 0. Dann arbeite die Kontrollturingmaschine T' nach folgender Vorschrift:

(1) Kopiere $d(N)$ auf Band 2; schreibe $d(0)$ auf Band 3.
(2) Falls Band 2 mit ‚0' endet, lösche dieses Zeichen; sonst HALT 2.
(3) Lies ein Zeichen $y_i \in \{0,1\}$ der Hilfseingabe.
(4) Falls $y_i = 1$, addiere die nächste Dualzahl $d(k_i)$ zum Inhalt von Band 3;
falls $y_i \neq 1$, überlese k_i;
falls die Eingabe zu Ende, gehe zu (5), sonst zu (3).
(5) Vergleiche die Inhalte von Band 2 und Band 3; falls gleich, HALT 1, sonst HALT 2.

Die Korrektheit der hierdurch gegebenen Maschine erhält man wie diejenige von T im Kurstext. Die Schleife (3) – (4) wird höchstens $n = \lg(x)$-mal durchlaufen, und jeder Durchlauf kostet höchstens $O(n)$ Rechenschritte. Die Schranke $O(n)$ wird auch bei der Ausführung der übrigen Befehle (1), (2) und (5) nicht überschritten. Damit folgt HALBER-RUCKSACK $\in$ NP.

(b) Es sei $\Sigma := \{0,1,\#\}$. Wir definieren $f : \Sigma^* \to \Sigma^*$ für alle $x \in \Sigma^*$ durch

$$f(x) := \begin{cases} d(2N)\#d(k_1)\dots\#d(k_n) & \text{falls } \begin{cases} \exists\, n \geq 1,\ N, k_1, \dots, k_n \in \mathbb{N}. \\ x = d(N)\#d(k_1)\dots\#d(k_n) \end{cases} \\ \varepsilon & \text{sonst.} \end{cases}$$

Dann gilt für alle $x \in \Sigma^*$:

$$\begin{aligned} x \in \text{RUCKSACK} &\iff \begin{cases} \exists\, n \geq 1,\ N, k_1, \dots, k_n \in \mathbb{N}. \\ x = d(N)\#d(k_1)\dots\#d(k_n) \wedge \\ \exists\, M \subseteq \{1,\dots,n\}.\ \sum_{i\in M} k_i = N \end{cases} \\ &\iff \begin{cases} \exists\, n \geq 1,\ N, k_1, \dots, k_n \in \mathbb{N}. \\ f(x) = d(2N)\#d(k_1)\dots\#d(k_n) \wedge \\ \exists\, M \subseteq \{1,\dots,n\}.\ \sum_{i\in M} k_i = N \end{cases} \\ &\iff \begin{cases} \exists\, n \geq 1,\ N' \text{ gerade}, k_1, \dots, k_n \in \mathbb{N}. \\ f(x) = d(N')\#d(k_1)\dots\#d(k_n) \wedge \\ \exists\, M \subseteq \{1,\dots,n\}.\ \sum_{i\in M} k_i = \frac{N'}{2} \end{cases} \\ &\iff f(x) \in \text{HALBER-RUCKSACK}. \end{aligned}$$

Dies zeigt

$$\text{RUCKSACK} \leq_{pol} \text{HALBER-RUCKSACK},$$

denn f ist in polynomieller Zeit berechenbar. (f testet die syntaktische Form der Eingabe und fügt eine ‚0' an $d(N)$ bzw. gibt ε aus.) Damit ist HALBER-RUCKSACK NP-vollständig.

Aufgabe 5 (a) Induktionsanfang $n = 0$:

Es sei $w \in \Delta^*$, und es gelte $\mathtt{S} \xrightarrow{0} w$. Dann folgt $w = \mathtt{S} = (\mathtt{ab})^0\mathtt{Sb}^0$. Damit gilt die Behauptung in diesem Falle.

Induktionsschluss $n \to n+1$:

Es sei wiederum $w \in \Delta^*$, und es gelte nun $\mathtt{S} \xrightarrow{n+1} w$. Dann gibt es nach Definition 7.1.6 ein Wort $v \in \Delta^*$ mit $\mathtt{S} \xrightarrow{n} v \xrightarrow{1} w$. Auf v ist die Induktionsvoraussetzung anwendbar. Danach gilt

$$v = \mathtt{a} \text{ oder } (n \geq 2 \text{ und}) \; v = (\mathtt{ab})^n\mathtt{b}^{n-2} \text{ oder } v = (\mathtt{ab})^n\mathtt{Sb}^n.$$

Aus $v \xrightarrow{1} w$ folgt, dass v kein Terminalwort ist. Damit können die ersten beiden Fälle nicht zutreffen. Es gilt somit $v = (\mathtt{ab})^n\mathtt{Sb}^n$.

Wir unterscheiden nun zwei Fälle.

1. Fall: Beim Übergang $v \xrightarrow{1} w$ wurde die Regel $\mathtt{S} \to \mathtt{abSb}$ angewendet. Dann erhält man $w = (\mathtt{ab})^{n+1}\mathtt{Sb}^{n+1}$. Also gilt die Behauptung hier.

2. Fall: Beim Übergang $v \xrightarrow{1} w$ wurde die Regel $\mathtt{S} \to \mathtt{a}$ angewendet. Für den Fall $n = 0$ ist die Gültigkeit der Behauptung dann offensichtlich. Andernfalls ergibt sich ($n + 1 \geq 2$ und) $w = (\mathtt{ab})^n\mathtt{ab}^n = (\mathtt{ab})^{n+1}\mathtt{b}^{n-1}$. Also gilt die Behauptung auch in diesem Fall.

Damit ist die Induktion beendet, und die Behauptung ist bewiesen.

(b) Es soll gezeigt werden, dass $L(G) \subseteq L$ gilt. Sei dazu $w \in L(G)$. Dann gilt nach Definition 7.1.6: Es gibt ein $n \in \mathbb{N}$ mit $\mathtt{S} \xrightarrow{n} w$, und $w \in \Sigma^*$. Nach der unter (a) gezeigten Behauptung folgt daraus $w = \mathtt{a}$ oder ($n \geq 2$ und) $w = (\mathtt{ab})^n\mathtt{b}^{n-2}$. Damit ist $w \in L$, w. z. z. w.

(c) Wir führen einen Widerspruchsbeweis und nehmen dazu an, L sei regulär. Dann gibt es nach dem Pumping–Lemma für reguläre Mengen (Satz 8.4.3) ein $n \in \mathbb{N}$, so dass für alle $t, z, \bar{t} \in \Sigma^*$ mit $tz\bar{t} \in L$ und $\lg(z) = n$ Wörter $u, v, w \in \Sigma^*$ derart existieren, dass $z = uvw$, $v \neq \varepsilon$ und $\forall i \geq 0 \,.\, tuv^iw\bar{t} \in L$ gilt.

Wir nehmen nun ein solches n und setzen $t := (\mathtt{ab})^{n+2}$, $z := \mathtt{b}^n$ und $\bar{t} := \varepsilon$. Dann ist offensichtlich $tz\bar{t} = (\mathtt{ab})^{n+2}\mathtt{b}^n \in L$.

Daher gibt es $u, v, w \in \Sigma^*$ mit den obigen Eigenschaften. Da v ein Teilwort von $z = \mathtt{b}^n$ ist, können wir schreiben:

$$v = \mathtt{b}^{\lg(v)} \text{ und } tuv^0w\bar{t} = (\mathtt{ab})^{n+2}\mathtt{b}^{n-\lg(v)}.$$

Da $\lg(v) > 0$ ist, endet das Wort mit weniger als $n+1$ b's nach dem letzten a. Da jedoch $n+2$ a's in dem Wort vorkommen, ergibt sich ein Widerspruch zu $tuv^0w\bar{t} \in L$. Also ist unsere Annahme falsch, d. h., L ist nicht regulär.

(d) Die Antwort ist „ja". Es ist nämlich plausibel (oder leicht mit Hilfe einer geeigneten Induktion zu zeigen), dass jedes $w \in L$ ableitbar ist, also $L \subseteq L(G)$ gilt. D. h. aber, dass $L = L(G)$ ist. Da G eine kontextfreie Grammatik darstellt, ist L somit kontextfrei.

Aufgabe 6 (a) Es ist

$$L(G) = \{\mathtt{a}^{2\cdot n+1}\mathtt{b}^{2\cdot m}\mathtt{a} \mid n, m \in \mathbb{N}\}.$$

(b) Wir führen noch einmal die Regeln der Menge R auf:

$$\begin{aligned}
&\mathtt{S} \longrightarrow \mathtt{aaS} \mid \mathtt{aT}\\
&\mathtt{T} \longrightarrow \mathtt{bbT} \mid \mathtt{U}\\
&\mathtt{U} \longrightarrow \mathtt{a}
\end{aligned}$$

Nun wenden das Verfahren aus Kurseinheit 5 an.

Normieren der Terminalregeln (NT) liefert:

$$\begin{aligned}
&\mathtt{S} \longrightarrow \mathtt{aaS} \mid \mathtt{aT}\\
&\mathtt{T} \longrightarrow \mathtt{bbT} \mid \mathtt{U}\\
&\mathtt{U} \longrightarrow \mathtt{aE}\\
&\mathtt{E} \longrightarrow \varepsilon
\end{aligned}$$

Verkürzung der Regellängen (VR), zweimal angewendet, liefert:

$$\begin{aligned}
&\mathtt{S} \longrightarrow \mathtt{aX} \mid \mathtt{aT}\\
&\mathtt{X} \longrightarrow \mathtt{aS}\\
&\mathtt{T} \longrightarrow \mathtt{bY} \mid \mathtt{U}\\
&\mathtt{Y} \longrightarrow \mathtt{bT}\\
&\mathtt{U} \longrightarrow \mathtt{aE}\\
&\mathtt{E} \longrightarrow \varepsilon
\end{aligned}$$

Elimination (EL) der einzigen längentreuen Regel $\mathtt{T} \longrightarrow \mathtt{U}$ liefert:

$$\begin{aligned}
\mathtt{S} &\longrightarrow \mathtt{aX} \mid \mathtt{aT} \\
\mathtt{X} &\longrightarrow \mathtt{aS} \\
\mathtt{T} &\longrightarrow \mathtt{bY} \mid \mathtt{aE} \\
\mathtt{Y} &\longrightarrow \mathtt{bT} \\
\mathtt{U} &\longrightarrow \mathtt{aE} \\
\mathtt{E} &\longrightarrow \varepsilon
\end{aligned}$$

Wir nennen diese Regelmenge R'. Die Grammatik

$$G' = (\{\mathtt{S}, \mathtt{T}, \mathtt{U}, \mathtt{X}, \mathtt{Y}, \mathtt{E}\}, \{\mathtt{a}, \mathtt{b}\}, R', \mathtt{S})$$

ist dann eine rechtslineare Normalformgrammatik mit $L(G') = L(G)$.

Aufgabe 7 (a) Die Grammatik $G = (\{\mathtt{S}, \mathtt{T}\}, \Sigma, R, \mathtt{S})$ mit der durch

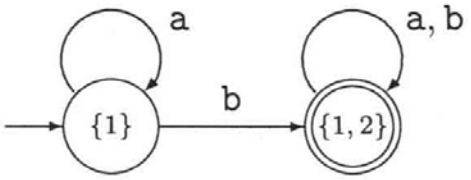
$$\begin{aligned}
\mathtt{S} &\longrightarrow \mathtt{aS} \mid \mathtt{bS} \mid \mathtt{bT} \\
\mathtt{T} &\longrightarrow \mathtt{aT} \mid \varepsilon
\end{aligned}$$

gegebenen Regelmenge R ist rechtslinear und in Normalform, und sie erfüllt $L(G) = L(A)$. (Siehe Beweis von Lemma 8.3.5 (2).)

(b) Wir wenden den Algorithmus 8.3.9 an und erhalten – analog zu Beispiel 8.3.10 – die folgende Tabelle:

P	a	b	$W \setminus V$
$\{1\}$	$\{1\}$	$\{1,2\}$	$\{1,2\}$
$\{1,2\}$	$\{1,2\}$	$\{1,2\}$	$\emptyset$

Daraus ergibt sich der äquivalente determinierte Automat, der so genannte Potenzautomat, graphisch wie folgt:



Aufgabe 8 1. (a) Falsch. ($L(G') = L(G) \setminus \{\varepsilon\}$.)

(b) Falsch. (Nicht *genau dann,* sondern nur *dann.*)

(c) Falsch. (Es gilt vielmehr die Umkehrung.)

(d) Falsch. (Auch hier müssen die Sprachklassen wieder vertauscht werden.)

(e) Falsch. (Nicht einmal überhaupt kontextfrei.)

2. (a) Richtig. (Satz 9.6.5.)

(b) Falsch. (Der Abschluss gilt nur für die Vereinigung; hinsichtlich des Durchschnitts ist im Kurstext ein Gegenbeispiel angeführt (s. Beweis von Korollar 9.6.6).)

(c) Falsch. (Für den Durchschnitt greift dasselbe Beispiel.)

(d) Richtig. (Satz 9.7.3.)

(e) Richtig. (Satz 9.7.3.)