Grundlagen der theoretischen Informatik, SS 2001

Nachklausur zum 2. Teil, Dienstag, 9. Oktober 2001 — Lösungen

Aufgabe	1	2	3	(4)
Punkte	10	10	10	(5)
Punkte				

gesamt:

Anleitung: Bearbeiten Sie jede Aufgabe auf dem dafür vorgesehenen *Blatt*. Verwenden Sie gegebenfalls Ergänzungsblätter. Schreiben Sie auf *alle* Blätter Ihren Namen und Ihre Matrikelnummer.
Sie können alle Aussagen, die in der Vorlesung oder auf den Übungsblättern vorgekommen sind, verwenden, ohne die Beweise abzuschreiben.
Kriterium für positive Bewertung: mindestens 50 %, das sind 15 Punkte.

---

1. (10 Punkte) Wandeln Sie die folgende Grammatik in Chomsky-Normalform um. Prüfen Sie dann mit dem CYK-Algorithmus nach, dass das Wort $aaa{+}aa{+}a{-}$ von dieser Grammatik erzeugt wird, und geben Sie einen Syntaxbaum *und* eine Linksableitung (für die umgewandelte Grammatik) an.

$$\begin{aligned} S &\to SSS \mid A \\ A &\to A{+}A \mid AS{-} \mid a \end{aligned}$$

Lösung: Erster Schritt. Trennung von Variablen und Terminalsymbolen:

$$S \to SSS \mid A, \qquad A \to APA \mid ASM \mid a, \qquad P \to +, \qquad M \to -$$

Zweiter Schritt. Elimination von zu langen Regeln:

$$\begin{aligned} &S \to ST,\ T \to SS; \quad S \to A \\ &A \to AB,\ B \to PA; \quad A \to AC,\ C \to SM \\ &P \to +,\ \ M \to -,\ \ A \to a \end{aligned}$$

Dritter Schritt. Elimination der zu kurzen Regel $S \to A$: Sie wird ersetzt durch $S \to AB$, $S \to AC$, $S \to a$.

Eine Linksableitung ist dann $S \to AC \to aC \to aSM \to aSTM \to aaTM \to aaSSM \to aaABSM \to aaaBSM \to aaaPASM \to aaa{+}ASM \to aaa{+}aSM \to aaa{+}aaM \to aaa{+}aa{-}$.

2. (10 Punkte) Beweisen Sie, dass die folgenden Sprachen über dem Alphabet $\{a, b, c\}$ nicht kontextfrei sind.

   (a) $L_1 = \{\, w \mid w$ enthält gleich viele a's, b's und c's. $\}$

   (b) $L_2 = \{\, a^i b^j c^k \mid 1 \le i < j = k \,\}$

Lösung zu (a): $L_1 \cap a^*b^*c^* = \{\, a^i b^i b^i \mid i \geq 0 \,\}$ und diese Sprache ist bekanntlich nicht kontextfrei. Beim Schnitt mit einer regulären Sprache bleibt eine kontextfreie Sprache kontextfrei.

Man kann auch das Pumping-Lemma auf $a^N b^N c^N \in L_1$ anwenden: $a^N b^N c^N = uvxyz$. Wegen $|vxy| \leq N$ kann vy nicht a's, b's und c's, enthalten. Daher ist $vy \in \{a,b\}^+ \cup \{b,c\}^+$. Somit enthält uxz nicht gleich viele a's, b's und c's. Widerspruch.

Lösung zu (b): Beweis durch Widerspruch. N sei die Konstante aus dem Pumping-Lemma. Wende das Pumping-Lemma auf $a^N b^{N+1} c^{N+1} \in L_2$ an: Dann ist $a^N b^{N+1} c^{N+1} = uvxyz$ mit $|vxy| \leq N$ und $|vy| \geq 1$, und alle Wörter uv^ixy^iz für $i \geq 0$ gehören zu L_2.

(a) Wenn v oder y nicht von der Form $a^* \cup b^* \cup c^*$ ist, dann wäre $uv^2xy^2z \notin a^*b^*c^*$ und somit $uv^2xy^2z \notin L_2$.

(b) Wenn $y = c^i$ $(i \geq 0)$ ist, muss $v = b^i$ sein, denn sonst hätte uxz nicht gleich viele b's und c's. Also ist $i \geq 1$. Dann hat aber uxz nicht mehr b's als a's und gehört somit nicht zu L_2, ein Widerspruch.

(c) Wenn $y = b^i$ $(i \geq 1)$ ist, dann ist $v \in a^* \cup b^*$, und uv^2xy^2z enthält mehr b's als c's.

(d) Wenn $y = a^i$ $(i \geq 1)$ ist, dann ist auch $v \in a^*$, und uv^ixy^iz enthält für genügend großes i mehr a's als b's.

Damit sind alle Möglichkeiten für y erschöpft.

3. (10 Punkte) Geben Sie für die folgende Grammatik einen äquivalenten Kellerautomaten an. (Entscheiden Sie selbst, wie der Automat akzeptieren soll.)

$$\begin{aligned} S &\to S - A \ \mid \ A * A \\ A &\to A - A \ \mid \ a \end{aligned}$$

Lösung: „Mechanische" Übersetzung der Grammatik. $Q = \{q_0\}$, $\Gamma = \{S, A, *, -, a\}$, $\Sigma = \{a, *, -\}$, $Z_0 = S$. Es gibt nur einen einzigen Zustand q_0; deshalb werden im Folgenden bei $\delta \subseteq Q \times \Gamma \times \Sigma \times Q \times \Gamma^*$ die Zustände nicht geschrieben.

$$\delta = \{(S, \varepsilon, A - S), (S, \varepsilon, A * A), (A, \varepsilon, A - A), (-, -, \varepsilon), (*, *, \varepsilon), (A, a, \varepsilon)\}.$$

(Achtung auf die Reihenfolge bei der ersten Regel.) Der Automat akzeptiert mit leerem Keller.

Man kann auch zur größeren Klarheit eigene Kellersymbole für $*$ und $-$ einführen, dann ist

$$\delta = \{(S, \varepsilon, AMS), (S, \varepsilon, ATA), (A, \varepsilon, AMA), (M, -, \varepsilon), (T, *, \varepsilon), (A, a, \varepsilon)\}.$$

Andere (ursprünglich nicht vorgesehene) Lösungsmöglichkeit: Die Grammatik erzeugt die reguläre Sprache $L = a(-a)^* * a(-a)^*$. Der Kellerautomat kann einen endlichen Automaten für diese Sprache simulieren.

4. (Zusatzfrage, 5 Punkte) Ist die Sprache

$$L = \{\, 0^i 1^j 0^k 1^l \mid i + l = j + k \,\}$$

kontextfrei? Ist sie regulär? Beweisen Sie Ihre Antworten.

Lösung: 1. Sie ist kontextfrei. Man kann $L = L_1 \cup L_2$ schreiben mit

$$L_1 = \{\, 0^i 1^j 0^k 1^l \mid i + l = j + k, i \leq j \,\}, \quad L_2 = \{\, 0^i 1^j 0^k 1^l \mid i + l = j + k, l \leq k \,\}$$

L_1 kann man mit $m = i - j$ so schreiben:

$$L_1 = \{\, 0^i 1^i 1^m 0^k 1^l \mid i \geq 0, m + k = l \,\} = \{\, 0^i 1^i \mid i \geq 0 \,\} \cdot \{\, 1^m 0^k 1^{m+k} \mid m, k \geq 0 \,\},$$

und analog für L_2. Diese Sprachen lassen sich einfach durch kontextfreie Grammatiken erzeugen. Die folgende Grammatik erzeugt L.

$$\begin{array}{lll}
S \to NA \mid AN & & \\
N \to 0N1 \mid \varepsilon & & (N \text{ erzeugt } \{\, 0^i 1^i \,\}.) \\
A \to 1A1 \mid N & & (A \text{ erzeugt } \{\, 1^i 0^j 1^{i+j} \,\}.) \\
B \to 0B0 \mid N & & (B \text{ erzeugt } \{\, 0^{i+j} 1^i 0^j \,\}.)
\end{array}$$

Ein anderer Beweis konstruiert einen (nichtdeterministischen) Kellerautomaten, der die Differenz zwischen den Häufigkeiten der Symbole zählt. Der Automat durchläuft 4 Phasen:

Phase 1 (Zustand q_0): Jede 0 wird als $+1$ gezählt.
Phase 2 (Zustand q_1): Jede 1 wird als -1 gezählt.
Phase 3 (Zustand q_2): Jede 0 wird als -1 gezählt.
Phase 4 (Zustand q_3): Jede 1 wird als $+1$ gezählt.

Der Automat akzeptiert, wenn der Zähler am Ende 0 ist. Ein Zähler kann durch eine Folge von $+$-Symbolen (für positive Werte) oder $-$-Symbolen (für negative Werte) auf dem Keller dargestellt werden. Ein „leerer“ Keller entspricht dem Wert 0. Der Übergang von einer Phase zur nächsten ist nichtdeterministisch, weil $j = 0$ oder $k = 0$ nicht ausgeschlossen ist.

Der folgende Automat akzeptiert L durch leeren Keller: $Q = \{q_0, q_1, q_2, q_3\}$, $\Gamma = \{Z_0, +, -\}$, und

$$\begin{aligned}
\delta = \{ \quad & (q_0, \gamma, 0, q_0, +\gamma), (q_0, \gamma, \varepsilon, q_1, \gamma), \\
& (q_1, +, 1, q_1, \varepsilon), (q_1, \gamma^-, 1, q_1, -\gamma^-), (q_1, \gamma, \varepsilon, q_2, \gamma), \\
& (q_2, +, 0, q_2, \varepsilon), (q_2, \gamma^-, 0, q_2, -\gamma^-), (q_2, \gamma, \varepsilon, q_3, \gamma), \\
& (q_3, -, 1, q_3, \varepsilon); \ (q_3, Z_0, \varepsilon, q_3, \varepsilon) \quad \},
\end{aligned}$$

für $\gamma \in \Gamma$ und $\gamma^- \in \{Z_0, -\}$.

2. Die Sprache L ist nicht regulär. Zum Beispiel ist $L \cap 010^+1^+ = \{\, 010^i 1^i \mid i \geq 1 \,\}$, und diese Sprache ist nicht regulär, ahnlich wie die bekannte nichtreguläre Sprache $\{\, 0^i 1^i \mid i \geq 1 \,\}$. (Vergleiche auch Aufgabe 4 der ersten Nachklausur zum ersten Teil.)

Man kann auch das Pumping-Lemma direkt anwenden.