Theoretische Informatik

Prof. Dr. Peter Thiemann
Marius Weidner
Simon Dorer

Universität Freiburg
Institut für Informatik
Sommersemester 2026

Übungsblatt 6 – Lösungen

Abgabe: Dienstag, 11.06.2026, 16:00 Uhr

Hinweis:
In diesem Übungsblatt fassen wir mehrere Produktionsregeln mit derselben linken Seite verkürzt zusammen. Statt

$$\begin{gathered} A \to \alpha_1 \\ A \to \alpha_2 \\ \vdots \\ A \to \alpha_n \end{gathered}$$

schreiben wir kurz $A \to \alpha_1 \mid \alpha_2 \mid \ldots \mid \alpha_n$.

Aufgabe 6.1 (Grammatiken; 5 Punkte)
Konstruieren Sie für jede der folgenden Sprachen eine Grammatik über dem Alphabet $\Sigma = \{a, b, c\}$. Erklären Sie dabei *kurz*, warum Ihre Grammatik genau die angegebene Sprache erzeugt.

(a) Geben Sie eine *reguläre* Grammatik $\mathcal{G}_{\text{reg}}$ an, sodass

$$L(\mathcal{G}_{\text{reg}}) = \{\, w \in \Sigma^* \mid w \text{ enthält nicht das Teilwort } aa \,\}.$$

(b) Geben Sie eine *kontextfreie* Grammatik $\mathcal{G}_{\text{cf}}$ an, sodass

$$L(\mathcal{G}_{\text{cf}}) = \{\, a^n b^m c^{n+m} \mid n, m \in \mathbb{N} \,\}.$$

Lösung:

(a) Eine mögliche reguläre Grammatik ist $\mathcal{G}_{\text{reg}} = (\Sigma, N, P, S)$, mit

$$N = \{S, A\} \quad \text{und} \quad P = \left\{ \begin{aligned} &S \to aA \mid bS \mid cS \mid \varepsilon, \\ &A \to bS \mid cS \mid \varepsilon \end{aligned} \right\}.$$

Das Nichtterminal A beschreibt dabei den Fall, dass das zuletzt erzeugte Zeichen ein a ist. Da aus A keine Produktion existiert, die ein weiteres a erzeugt, kann das Teilwort aa nicht auftreten. Umgekehrt kann jedes Wort über Σ, das aa nicht enthält, erzeugt werden: Nach jedem b oder c befindet sich die Grammatik wieder im Zustand S, aus dem erneut ein a erzeugt werden darf.

(b) Eine mögliche kontextfreie Grammatik ist $\mathcal{G}_{\text{cf}} = (\Sigma, N, P, S)$ wobei

$$N = \{S, B\} \quad \text{und} \quad P = \left\{ \begin{array}{l} S \to aSc \mid B, \\ B \to bBc \mid \varepsilon \end{array} \right\}.$$

Jede Anwendung der Produktion $S \to aSc$ erzeugt ein a am Anfang und ein zugehöriges c am Ende. Nach dem Wechsel $S \to B$ erzeugt jede Anwendung der Produktion $B \to bBc$ ein b und ein weiteres zugehöriges c. Beide Produktionen können beliebig oft, insbesondere auch keinmal, angewendet werden. Daher sind auch Wörter ohne a, ohne b sowie das leere Wort ε enthalten.

Aufgabe 6.2 (Grammatiken und starke Induktion; 5 Punkte)
Gegeben sei die kontextfreie Grammatik $\mathcal{G} = (\Sigma, N, P, S)$ mit

$$\begin{aligned} \Sigma &= \{0, 1\}, \\ N &= \{S\}, \\ P &= \{S \to 1S0S \mid 0S1S \mid \varepsilon\} \end{aligned}$$

und eine Sprache $L = \{\, w \in \Sigma^* \mid \#_0(w) = \#_1(w) \,\}$, wobei $\#_a(w)$ die Anzahl der Vorkommen des Zeichens $a \in \Sigma$ im Wort $w \in \Sigma^*$ bezeichnet.

Zeigen Sie mittels *starker Induktion*[1] über der Länge $|w|$ eines Wortes $w \in L$, dass

$$L \subseteq \mathcal{L}(\mathcal{G}).$$

Verwenden Sie dabei die Hilfsfunktion $d\colon \Sigma^* \to \mathbb{Z}$ mit

$$d(\varepsilon) = 0, \qquad d(1w) = d(w) + 1, \qquad d(0w) = d(w) - 1$$

für alle $w \in \Sigma^*$, sowie die Eigenschaften

$$u \in L \iff d(u) = 0 \qquad \text{und} \qquad d(uv) = d(u) + d(v)$$

für alle $u, v \in \Sigma^*$.

Lösung:
Zunächst gilt für alle $u \in \Sigma^*$, dass $u \in L \iff d(u) = 0$. Wir zeigen nun $L \subseteq \mathcal{L}(\mathcal{G})$ per starker Induktion über die Länge des Wortes.

IA) Sei $n = 0$. Dann ist $w = \varepsilon$. Wegen der Produktion $S \to \varepsilon$ gilt $S \vdash_{\mathcal{G}} \varepsilon$, also $\varepsilon \in \mathcal{L}(\mathcal{G})$.

IV) Sei $n > 0$. Für alle Wörter $u \in L$ mit $|u| < n$ gelte bereits $u \in \mathcal{L}(\mathcal{G})$.

IS) Sei $w \in L$ mit $|w| = n$. Dann ist $d(w) = 0$. Da $n > 0$, gibt es $a \in \{0, 1\}$ und $w' \in \Sigma^*$ mit $w = aw'$.

[1] Bei starker Induktion, wird in der Induktionsvoraussetzung angenommen, dass die Aussage für *alle* $n' < n$ gilt. Siehe hierzu https://de.wikipedia.org/wiki/VollstÃďndige_Induktion#Starke_Induktion.

Fall $a = 0$: Dann gilt

$$0 = d(w) = d(0w') = d(w') - 1,$$

also $d(w') = 1$. Wir zerlegen w' nun in $u1v$ mit $d(u) = d(v) = 0$. Schreibe $w' = a_1 \dots a_m$ und betrachte die Präfixwerte

$$d_i = d(a_1 \dots a_i) \qquad (0 \leq i \leq m),$$

wobei $d_0 = 0$. Wegen $d_m = d(w') = 1$ existiert ein kleinstes $k \geq 1$ mit $d_k = 1$. Für $u = a_1 \dots a_{k-1}$ gilt dann $d(u) = d_{k-1} = 0$, und wegen $d_k - d_{k-1} = 1$ ist $a_k = 1$. Also

$$w' = u1v \quad \text{mit} \quad d(u) = 0.$$

Aus $d(w') = d(u1v) = d(u) + 1 + d(v) = 1$ folgt außerdem $d(v) = 0$. Damit sind $u, v \in L$. Wegen $|u|, |v| < n$ liefert die Induktionsvoraussetzung

$$S \vdash_{\mathcal{G}}^* u \quad \text{und} \quad S \vdash_{\mathcal{G}}^* v.$$

Daraus erhalten wir die Ableitung

$$S \vdash_{\mathcal{G}} 0S1S \vdash_{\mathcal{G}}^* 0u1S \vdash_{\mathcal{G}}^* 0u1v = w.$$

Also gilt $w \in \mathcal{L}(\mathcal{G})$.

Fall $a = 1$: Analog gilt $d(w') = -1$. Sei k die erste Position in w', an der der Präfixwert -1 erreicht wird. Dann lässt sich w' als $u0v$ mit $d(u) = d(v) = 0$ schreiben. Nach Induktionsvoraussetzung gilt $S \vdash_{\mathcal{G}}^* u$ und $S \vdash_{\mathcal{G}}^* v$, also

$$S \vdash_{\mathcal{G}} 1S0S \vdash_{\mathcal{G}}^* 1u0S \vdash_{\mathcal{G}}^* 1u0v = w.$$

Damit ist auch in diesem Fall $w \in \mathcal{L}(\mathcal{G})$.

Nach starker Induktion folgt $L \subseteq \mathcal{L}(\mathcal{G})$.

$\square$

Aufgabe 6.3 (Ableitungsbäume; 5 Punkte)
Gegeben sei die kontextfreie Grammatik $\mathcal{G} = (\Sigma, N, P, S)$ mit

$$\begin{aligned}
\Sigma &= \{x, y, +\}, \\
N &= \{S\}, \\
P &= \{S \to S + S \mid x \mid y\}
\end{aligned}$$

(a) Zeigen Sie, dass $\mathcal{G}$ nicht eindeutig ist.

(b) Geben Sie eine eindeutige Grammatik $\mathcal{G}'$ an mit $L(\mathcal{G}') = L(\mathcal{G})$. Begründen Sie *kurz* warum Ihre Grammatik eindeutig ist und die gewünschte Sprache erzeugt.

Lösung:

(a) Wir zeigen, dass $\mathcal{G}$ nicht eindeutig ist, per Widerspruch. Angenommen, $\mathcal{G}$ ist eindeutig. Dann besitzt jedes Wort $w \in L(\mathcal{G})$ genau einen Ableitungsbaum.

Das Wort $x+y+x$ besitzt jedoch die folgenden zwei verschiedenen Ableitungsbäume:

$$\begin{array}{ccccc}
 & & S \to S+S & & \\
 & \diagup & & \diagdown & \\
 & S \to S+S & & & S \to x \\
\diagup & & \diagdown & & \\
S \to x & & S \to y & &
\end{array}
\qquad
\begin{array}{ccccc}
 & & S \to S+S & & \\
 & \diagup & & \diagdown & \\
S \to x & & & S \to S+S & \\
 & & \diagup & & \diagdown \\
 & & S \to y & & S \to x
\end{array}$$

Der linke Baum entspricht der Klammerung $(x + y) + x$, der rechte Baum der Klammerung $x + (y + x)$. Beide Ableitungsbäume haben also dasselbe abgeleitete Wort $x + y + x$. Dies ist ein Widerspruch zur Eindeutigkeit von $\mathcal{G}$. Somit ist $\mathcal{G}$ nicht eindeutig.

$\square$

(b) Eine eindeutige Grammatik für dieselbe Sprache ist $\mathcal{G}' = (\Sigma, N', P', E)$ mit

$$N' = \{E, R, A\} \quad \text{und} \quad P' = \left\{ \begin{array}{l} E \to AR, \\ R \to +AR \mid \varepsilon, \\ A \to x \mid y \end{array} \right\}.$$

Diese Grammatik erzeugt genau die Wörter $w \in \{x, y\}(\{+\}\{x, y\})^*$, also genau nichtleere Folgen von x- und y-Symbolen, die durch $+$ getrennt sind. Das ist auch die von $\mathcal{G}$ erzeugte Sprache: Jede Anwendung von $S \to S + S$ verbindet zwei solche nichtleeren Folgen mit einem $+$, und umgekehrt kann jede solche Folge in $\mathcal{G}$ induktiv erzeugt werden.

Die Grammatik $\mathcal{G}'$ ist eindeutig, da E immer genau ein erstes Atom A erzeugt und R danach deterministisch entscheidet: Bei einem folgenden $+$ muss $R \to +AR$ verwendet werden, am Wortende muss $R \to \varepsilon$ verwendet werden. Für A ist durch das nächste Terminal eindeutig festgelegt, ob $A \to x$ oder $A \to y$ verwendet wird.

Aufgabe 6.4 (Sep und Bin; 5 Punkte)
Betrachten Sie die Grammatik $\mathcal{G} = (\Sigma, N, P, S)$ mit

$$\Sigma = \{a, b, c\}, \quad N = \{S, A, B, C, D\} \quad \text{und}$$

$$P = \left\{ \begin{array}{l} S \to aBCb \mid ABc \mid D, \\ A \to BaC \mid b, \\ B \to Cb \mid \varepsilon, \\ C \to cSA \mid aD, \\ D \to AC \mid c \end{array} \right\}$$

(a) Wenden Sie den Algorithmus SEP aus der Vorlesung auf $\mathcal{G}$ an.

(b) Wenden Sie anschließend den Algorithmus BIN auf die in Teil (a) erhaltene Grammatik an. Führen Sie neue Nichtterminale systematisch ein und geben Sie die resultierende Grammatik vollständig an.

Lösung:

(a) Nach Anwendung von SEP erhalten wir $\mathcal{G}_{\text{SEP}} = (\Sigma, N_{\text{SEP}}, P_{\text{SEP}}, S)$ mit

$$N_{\text{SEP}} = \{S, A, B, C, D, Y_a, Y_b, Y_c\}$$

und

$$P_{\text{SEP}} = \left\{ \begin{array}{l} S \to Y_aBCY_b \mid ABY_c \mid D, \\ A \to BY_aC \mid b, \\ B \to CY_b \mid \varepsilon, \\ C \to Y_cSA \mid Y_aD, \\ D \to AC \mid c, \\ Y_a \to a, \\ Y_b \to b, \\ Y_c \to c \end{array} \right\}.$$

(b) Nun wenden wir BIN auf alle rechten Seiten der Länge größer als 2 an. Wir binarisieren dabei von links nach rechts, also zum Beispiel

$$S \to Y_aBCY_b$$

durch

$$S \to Y_aT_1, \qquad T_1 \to BT_2, \qquad T_2 \to CY_b.$$

Insgesamt erhalten wir $\mathcal{G}_{\text{BIN}} = (\Sigma, N_{\text{BIN}}, P_{\text{BIN}}, S)$ mit

$$N_{\text{BIN}} = \{S, A, B, C, D, Y_a, Y_b, Y_c, T_1, T_2, T_3, T_4, T_5\}$$

und

$$P_{\textsc{Bin}} = \left\{ \begin{array}{l} S \to Y_aT_1 \mid AT_3 \mid D, \\ T_1 \to BT_2, \\ T_2 \to CY_b, \\ T_3 \to BY_c, \\ A \to BT_4 \mid b, \\ T_4 \to Y_aC, \\ B \to CY_b \mid \varepsilon, \\ C \to Y_cT_5 \mid Y_aD, \\ T_5 \to SA, \\ D \to AC \mid c, \\ Y_a \to a, \\ Y_b \to b, \\ Y_c \to c \end{array} \right\}.$$

Aufgabe 6.5 (Erfahrungen; Datei: `NOTES.md`)

Notieren Sie Ihre Erfahrungen mit diesem Übungsblatt (benötigter Zeitaufwand, Probleme, Bezug zur Vorlesung, Interessantes, etc.).

Editieren Sie hierzu die Datei `NOTES.md` im Abgabeordner dieses Übungsblattes auf unserer Webplattform. Halten Sie sich an das dort vorgegebene Format, da wir den Zeitbedarf automatisch statistisch auswerten. Die Zeitangabe `4.5 h` steht dabei für 4 Stunden und 30 Minuten.