Technische Universität München **Sommer 2016**
Prof. J. Esparza / Dr. M. Luttenberger, S. Sickert **10. Juni 2016**

Einführung in die theoretische Informatik – Aufgabenblatt 8

Beachten Sie: Soweit nicht explizit angegeben, sind Ergebnisse stets zu begründen!

Hausaufgaben: Abgabe bis zum 15.06.2016 (Mittwoch) um 12:00

Aufgabe 8.1 CYK-Algorithmus 2P+2P

Wir betrachten die Grammatik $G = (\{S, T, U, A, B, C\}, \{a, b, c\}, P, S)$ in CNF mit den folgenden Produktionen P:

$$
\begin{array}{ll}
S \to TS \mid CT \mid a & A \to a \\
T \to AU \mid TT \mid c & B \to b \\
U \to SB \mid AB & C \to c
\end{array}
$$

(a) Bestimmen Sie mit dem CYK-Algorithmus, ob $ccaab \in L(G)$ und $aabcc \in L(G)$. Geben Sie dabei auch die berechneten Tabellen an.

(b) Beschreiben Sie eine Erweiterung des CYK-Algorithmus, mit welcher für ein gegebens $w \in L(G)$ alle Ableitungsbäume bzgl. G berechnet werden können, und wenden Sie dieses Verfahren auf die Wörter aus (a) an.

Aufgabe 8.2 Infix-Abschluss 3P+1P

Wir betrachten eine Zerlegung $z = uvw$ für das Wort z. Man bezeichnet u dann als *Präfix*, w als *Suffix* und v als *Infix*. Der Infixabschluss einer Sprache L ist dann definiert als $L_{\text{infix}} = \{v \mid \exists u, w \in \Sigma^*.\ uvw \in L\}$.

(a) Geben Sie eine allgemeine Konstruktion an, die aus einer CFG G in CNF mit $L = L(G)$ eine Grammatik G' mit $L(G') = L_{\text{infix}}$ erzeugt.

(b) Wenden Sie Ihr Verfahren auf die Grammatik aus HA8.1 an.

Aufgabe 8.3 Pushdown Automaten / Kellerautomaten 2P+2P+2P

Konstruieren Sie für die folgenden Sprachen jeweils einen Kellerautomaten. Der Automat soll mit **leerem Stack** akzeptieren. Geben Sie zusätzlich für jeden Automaten jeweils ein nicht-leeres Wort w mit akzeptierendem Lauf an.

(a) $L_1 = \{a^n b^{3n} \mid n \geq 0\}$

(b) $L_2 = \{a^n b^m \in \{a, b\}^* \mid n \leq m \leq 2n\}$

(c) $L_3 = \{w \in \{a, b\}^* \mid 2\,|w|_a = 3\,|w|_b\}$

Aufgabe 8.4 Ogdens Lemma 2P

Zeigen Sie mit Hilfe des Lemmas von Ogden (TA7.3), dass die folgende Sprache nicht kontextfrei ist:

$$L = \{a^i b^j c^j \mid i \neq j\}$$

Tutoraufgaben: Besprechung in KW24

Erinnerung: Wir bezeichnen mit $L_\varepsilon(A)$, die Sprache die von einem PDA A mit leerem Stack akzeptiert wird. Weiterhin bezeichnen wir mit $L_F(A)$, die Sprache die von einem PDA A mit Endzuständen akzeptiert wird.

Notation von PDA-Regeln: Anstatt der in den Folien verwendeten Schreibweise $(q, YZ) \in \delta(p, a, X)$ für die Ersetzungsregeln eines PDA, schreibt man alternativ $pX \xrightarrow{a} qYZ$ $(p, q \in Q,\ X, Y, Z \in \Gamma,\ a \in \Sigma)$ oder stellt diese entsprechend als Graph mit Knotenmenge $Q\Gamma^{\leq 2}$ dar, wobei die Kante (pX, qYZ) dann mit a beschriftet ist.

Für den PDA

$$\delta(p, a, \perp) = \{(p, X\perp)\} \qquad \delta(p, a, X) = \{(p, XX)\} \qquad \delta(p, b, X) = \{(p, \varepsilon)\} \qquad \delta(p, b, \perp) = \{(p, \varepsilon)\}$$

schreibt man daher alternativ:

$$p\perp \xrightarrow{a} pX\perp \qquad pX \xrightarrow{a} pXX \qquad pX \xrightarrow{b} p \qquad p\perp \xrightarrow{\varepsilon} p$$

oder der stellt diesen entsprechend als Graph mit Knotenmenge Q dar, wobei die Kante (p, q) dann mit "$a, X/YZ$" beschriftet ist (siehe *Hopcroft at al. „Introduction to Automata Theory", Kapitel 6*):

$$\begin{array}{c} a, \perp/X\perp \\ \circlearrowleft \\ b, X/\varepsilon \circlearrowleft\ p\ \circlearrowright a, X/XX \\ \circlearrowright \\ \varepsilon, \perp/\varepsilon \end{array}$$

Aufgabe 8.1 Deterministische PDAs

In der Vorlesung haben Sie Sie Lemma 3.65 ohne Beweis gesehen:

Sei $L \subseteq \Sigma^*$. Dann sind äquivalent:

(a) Es gibt einen DPDA D mit $L_\varepsilon(D) = L$

(b) Es gibt einen DPDA D' mit $L_F(D') = L$ **und** kein Wort aus L ist ein echter Präfix von einem anderen Wort aus L.

Zeigen Sie diese Äquivalenz.

Aufgabe 8.2 Boolesche Ausdrücke

Wir betrachten folgende Grammatik für boolesche Ausdrücke über den *booleschen Variablen* x, y (welche somit Terminale der Grammatik sind):

$$S \to (S \wedge S) \mid (S \vee S) \mid \neg S \mid x \mid y$$

Konstruieren Sie einen DPDA D mit $L(G) = L_\varepsilon(D)$.

Aufgabe 8.3 CFG $\longleftrightarrow$ PDA

Wir üben die Übersetzung zwischen CFG und PDA:

(a) Überführen Sie folgende CFG G (Startsymbol S) zunächst in CNF[1] und dann in einen PDA A mit $L_\varepsilon(A) = L(G) \setminus \{\varepsilon\}$:

$$S \to SS \mid aSb \mid bSa \mid \varepsilon$$

(b) Übersetzen Sie folgenden PDA A (Startkonfiguration qX) in eine CFG G mit $L_\varepsilon(A) = L(G)$:

$$qX \xrightarrow{l} qX[YXZ] \qquad q[YXZ] \xrightarrow{\varepsilon} pY[XZ] \qquad q[XZ] \xrightarrow{\varepsilon} qXZ \qquad qX \xrightarrow{n} qX \qquad qX \xrightarrow{x,y} q\varepsilon \qquad pY \xrightarrow{a,o} q\varepsilon \qquad qZ \xrightarrow{r} q\varepsilon$$

Notation: $pX \xrightarrow{a} qYZ$ steht kurz für $(q, YZ) \in \delta(p, a, X)$.

[1] Wenden Sie die Regeln in der Reihenfolge (3) (4) (1) (2) an, um eine kompaktere Grammatik zu erhalten.