Prof. J. Esparza
S. Sickert, J. Krämer

Technische Universität München
KEINE ABGABE

Einführung in die theoretische Informatik

Sommersemester 2017 – Übungsblatt 8

Übungsblatt

Wir unterscheiden zwischen Übungs- und Abgabeblättern. Auf diesem *Übungsblatt* finden Sie eine Übersicht über die Kernaspekte, die Sie in Kalenderwoche 25 in den Tutorien diskutieren, üben und vertiefen. Die Aufgaben auf diesem Blatt dienen dem Üben und Verstehen des Vorlesungsstoffes, sowie dem *eigenständigen Erarbeiten* der Kernaspekte. Außerdem sollen Ihnen diese Aufgaben auch helfen, ein Gefühl dafür zu bekommen, was Sie inhaltlich in der Klausur erwartet. Klausuraufgaben können jedoch deutlich von den hier gestellten Aufgaben abweichen. Abschreiben und Auswendiglernen von Lösungen wird Ihnen daher keinen dauerhaften Erfolg in der Vorlesung bringen. Fragen zu den Übungsblättern können Sie montags bis donnerstags von 12 Uhr bis 14 Uhr in der *THEO-Sprechstunde* in Raum 03.11.034 stellen.

Kernaspekte

K8.1 korrektes Wiedergeben der folgenden Definitionen und Algorithmen

- PDA
- CYK-Algorithmus
- unterstes Kellerzeichen Z_0
- Stackalphabet Γ.

K8.2 zu einer gegebenen Sprache L einen PDA A angeben, so dass $\mathsf{L} = \mathsf{L}(\mathsf{A})$
K8.3 einen PDA A in allen verschiedenen Darstellungsformen angeben
K8.4 mithilfe des CYK-Algorithmus entscheiden, ob ein Wort w von einer Grammatik G erzeugt wird
K8.5 mithilfe des erweiterten CYK-Algorithmus alle Ableitungsbäume für ein Wort w bezüglich einer Grammmatik G angeben
K8.6 begründet entscheiden, ob gegebene Beispiele neu eingeführte Definitionen erfüllen
K8.7 Aussagen mit neu eingeführten Definitionen beweisen oder widerlegen

Notation

Erinnerung: Wir bezeichnen mit $\mathsf{L}_\varepsilon(A)$ die Sprache, die von einem PDA A mit leerem Stack akzeptiert wird. Weiterhin bezeichnen wir mit $\mathsf{L}_\mathsf{F}(\mathsf{A})$ die Sprache, die von einem PDA A mit Endzuständen F akzeptiert wird.

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

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

schreibt man daher alternativ:

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

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

$$\mathsf{a}, \perp/X\perp$$
$$\mathsf{b}, X/\varepsilon \circlearrowleft p \circlearrowright \mathsf{a}, X/XX$$
$$\varepsilon, \perp/\varepsilon$$

AUFGABE 8.1. (*Pushdown-Automata / Kellerautomaten*) Stufe B/C

Geben Sie für die folgenden Sprachen jeweils einen Kellerautomaten A_i in allen oben aufgeführten Darstellungsarten an, so dass $\mathsf{L}_i = \mathsf{L}(\mathsf{A}_i)$. Der Automat soll mit **leerem Stack** akzeptieren. Geben Sie dann zusätzlich für jeden Automaten jeweils ein nicht-leeres Wort w mit akzeptierendem Lauf an.

(a) $\mathsf{L}_1 = \{\mathsf{a}^n\mathsf{b}^{3n} \mid n \geq 0\}$
(b) $\mathsf{L}_2 = \{\mathsf{a}^n\mathsf{b}^m \in \{\mathsf{a},\mathsf{b}\}^* \mid n \leq m \leq 2n\}$
(c) $\mathsf{L}_3 = \{w \in \{\mathsf{a},\mathsf{b}\}^* \mid 2 \cdot |w|_{\mathsf{a}} = 3 \cdot |w|_{\mathsf{b}}\}$

AUFGABE 8.2. (*Komplemente von CFL*) Stufe C

Aus der Vorlesung wissen Sie bereits, dass kontextfreie Sprachen *nicht* über Schnitt und Komplement abgeschlossen sind. In dieser Aufgabe geht es darum zu sehen, dass es dennoch kontextfreie, nicht-reguläre Sprachen gibt, deren Komplement ebenfalls kontextfrei und nicht regulär ist. Sei $\mathsf{L} = \{\mathsf{a}^n\mathsf{b}^n\mathsf{c}^m \mid n, m \in \mathbb{N}_0\}$ über dem Alphabet $\Sigma = \{\mathsf{a},\mathsf{b},\mathsf{c}\}$ eine nicht-reguläre, kontextfreie Sprache.

(a) Geben Sie eine kontextfreie Grammatik G für L und eine für $\overline{\mathsf{L}}$ an.
(b) Geben Sie einen PDA für L und für $\overline{\mathsf{L}}$ an.

AUFGABE 8.3. (*CYK-Algorithmus*) Stufe C

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

$$\begin{aligned} \mathsf{S} &\to T\mathsf{S} \mid CT \mid \mathsf{a} & \qquad A &\to \mathsf{a} \\ T &\to AU \mid TT \mid \mathsf{c} & B &\to \mathsf{b} \\ U &\to \mathsf{S}B \mid AB & C &\to \mathsf{c} \end{aligned}$$

(a) Bestimmen Sie mit dem CYK-Algorithmus, ob $\mathsf{ccaab} \in \mathsf{L}(\mathsf{G})$ und $\mathsf{aabcc} \in \mathsf{L}(\mathsf{G})$. Geben Sie dabei auch die berechneten Tabellen an.
(b) Beschreiben Sie eine Erweiterung des CYK-Algorithmus, mit welcher für ein gegebenes $w \in \mathsf{L}(\mathsf{G})$ alle Ableitungsbäume bzgl. G berechnet werden können, und wenden Sie dieses Verfahren auf die Wörter aus (a) an.

AUFGABE 8.4. (*Abschlusseigenschaften kontextfreier Sprachen*) Stufe D

Beweisen Sie die folgenden beiden Aussagen:

(a) Der Präfixabschluss $\mathsf{L}_{pre} \coloneqq \{u \mid \exists v.\ uv \in \mathsf{L}\}$ einer kontextfreien Sprache $\mathsf{L} \subseteq \Sigma^*$ über einem Alphabet Σ ist wieder kontextfrei. Geben Sie hierzu ein Übersetzung an, die die Grammatik G für L in eine Grammatik G' umwandelt, sodass $\mathsf{L}_{pre} = \mathsf{L}(G')$ übersetzt.
(b) Kontextfreie Sprachen sind unter Schnitt mit regulären Sprachen abgeschlossen.

AUFGABE 8.5. (*CYK-Algorithmus (Teil 2)*) Stufe C

Sei G durch folgende Produktionen gegeben:

$$\mathsf{S} \to AB \mid CD \mid AT \mid CU \mid \mathsf{SS} \qquad T \to \mathsf{S}B \qquad U \to \mathsf{S}D \qquad A \to (\qquad B \to) \qquad C \to \{ \qquad D \to \}$$

(a) Wenden Sie den CYK-Algorithmus auf folgende Wörter an:

$$w_1 = ()\{()\} \qquad w_2 = ()\{(\qquad w_3 = ()\},$$

indem Sie die Tabellen, die Sie mithilfe des CYK-Algorithmus berechnen, angeben.
(b) Diskutieren Sie, wie man beim CYK-Algorithmus das mehrmalige Berechnen der gleichen Tabelle verhindern kann. Bestimmen Sie dazu, in welcher Beziehung die Worte w_1 bis w_3 zueinander stehen und warum es in Aufgabenteil (a) reicht, eine einzige Tabelle zu berechnen.
(c) Wir möchten den CYK-Algorithmus so abändern, dass er Eingabefehler (z.B. eine öffnende Klammer ohne passende schließende Klammer) erkennt und die Position des Fehlers bestimmt. *Geben Sie hierzu eine neue Variante des CYK-Algorithmus so an, dass er für gegebenes $w \in \Sigma^*$ den längsten Präfix u bestimmt, so dass sich u zu einem Wort in $\mathsf{L}(\mathsf{G})$ ergänzen lässt.*
Beispiel:

$$\mathsf{G}\colon \mathsf{S} \to AB \mid \mathsf{SS} \qquad A \to \mathsf{a} \qquad B \to \mathsf{b}$$

Dann gilt $w = \mathsf{ababb} \notin \mathsf{L}(\mathsf{G})$, aber $\mathsf{ab}, \mathsf{abab} \in \mathsf{L}(\mathsf{G})$; weiterhin gibt es kein $v \in \Sigma^*$, so dass $wv \in \mathsf{L}(\mathsf{G})$ gilt, womit abab das gesuchte Präfix ist. Im Fall $w = \mathsf{aba}$ wäre aba das gesuchte Präfix, da $w\mathsf{b} \in \mathsf{L}(\mathsf{G})$ gilt.
(d) Wenden Sie den Algorithmus aus (c) auf die Wörter aus (a) an, um das maximalen Präfix zu bestimmen, der sich noch zu einem Wort in $\mathsf{L}(\mathsf{G})$ ergänzen lässt.

AUFGABE 8.6. (*Ogdens Lemma*) Stufe E

Sei L eine CFL. Dann existiert ein $p \in \mathbb{N}_0$, so dass für jedes $z \in \mathsf{L}$ mit $|z| \geq p$ und jede *Markierung* von mindestens p Zeichen in z gilt: Es gibt eine Zerlegung $z = uvwxy$ mit

- vx enthält mindestens ein markiertes Zeichen.
- vwx enthält höchstens p markierte Zeichen.
- $uv^iwx^iy \in \mathsf{L}$ für jedes $i \in \mathbb{N}_0$.

Eine geeignete Markierung kann z.B. Unterstreichung $\underline{\mathsf{a}}$, Apostroph a' oder Einfärbung a sein. Markiert man stets alle Zeichen, so erhält man das ursprüngliche Pumping-Lemma für CFL.

(a) Sei $\mathsf{L} = \{a^ib^jc^kd^l \mid i = 0 \vee j = k = l\}$.
Zeigen Sie mittels Ogdens Lemma, dass L nicht kontextfrei ist.
(Warum reicht das Pumping-Lemma für CFL nicht aus, um zu zeigen, dass L nicht kontextfrei ist?)

(b) Beweisen Sie Ogdens Lemma.