Technische Universität München
Fakultät für Informatik
Prof. Tobias Nipkow, Ph.D.
Dr. Werner Meixner, Alexander Krauss

Sommersemester 2008
Übungsblatt 7
4. Juni 2008

Einführung in die Theoretische Informatik

Abgabetermin: 9. Juni 2008 **vor** *der Vorlesung in die THEO Briefkästen*

Hausaufgabe 1 (5 Punkte)

Eine Grammatik $G = (V, \Sigma, P, S)$ sei durch folgende Produktionen gegeben.

$$\begin{aligned} S &\rightarrow BAa\,, \\ A &\rightarrow Sa \mid Sb \mid a\,, \\ B &\rightarrow Bb \mid Sb \mid b\,. \end{aligned}$$

Konstruieren Sie mit entsprechenden Standardverfahren aus der Vorlesung einen PDA K, der $L(G)$ durch Endzustände akzeptiert.

Hausaufgabe 2 (5 Punkte)

Wir betrachten den Kellerautomaten $M = (\{q_0, q_1\}, \{a, b\}, \{A, B, Z_0\}, q_0, Z_0, \delta)$, der die Sprache $L_\epsilon(M)$ mit leerem Keller akzeptiert. Dabei sei δ gegeben durch

$$\begin{aligned} \delta(q_0, a, *) &= \{(q_0, A*)\}\,, \\ \delta(q_0, b, *) &= \{(q_0, B*)\}\,, \\ \delta(q_1, a, A) &= \{(q_1, \epsilon)\}\,, \\ \delta(q_1, b, B) &= \{(q_1, \epsilon)\}\,, \\ \delta(q_0, \epsilon, *) &= \{(q_1, *)\}\,, \\ \delta(q_1, \epsilon, Z_0) &= \{(q_1, \epsilon)\}\,. \end{aligned}$$

$*$ symbolisiert ein Kellerzeichen aus $\{A, B, Z_0\}$.

1. Konstruieren Sie mit Standardverfahren aus der Vorlesung eine kontextfreie Grammatik G, die $L_\epsilon(M)$ erzeugt.

2. Vereinfachen Sie die erhaltene Grammatik insbesondere durch Entfernung überflüssiger Variablen und geben Sie eine knappe informelle Beschreibung der erzeugten Sprache an.

Hausaufgabe 3 (5 Punkte)

Seien $L_1, L_2 \subseteq \Sigma^*$. Zeigen Sie:
Wenn L_1 kontextfrei ist und L_2 regulär, dann ist $L_1 \cap L_2$ kontextfrei.
Hinweis: Konstruieren sie aus einem PDA und einem DFA/NFA einen neuen PDA.

***Hinweis:** Die als* Vorbereitung *bezeichneten Aufgaben werden nicht bewertet und dienen der häuslichen Vorbereitung der Tutoraufgaben, die ebenfalls nicht bewertet werden. Die Abgabe einer Bearbeitung der Vorbereitungsaufgaben zusammen mit der Bearbeitung der Hausaufgaben wird empfohlen. Tutoraufgaben werden in den Übungsgruppen bearbeitet.*

Vorbereitung 1

Geben Sie einen DPDA an, der die Sprache $L = \{a^n b^{2n} \mid n \geq 1\}$ mit Endzustand akzeptiert. Erfüllt L die Präfixbedingung?

Tutoraufgabe 1

Sei M ein DPDA, so dass $L := L_F(M)$ die Präfixbedingung erfüllt. Konstruieren Sie einen DPDA M', mit $L_\epsilon(M') = L(M)$.

Vorbereitung 2

Die Auswertung von Polynomen $p(x) = \sum_{i=0}^{n} a_i x^i$ vom Grad $n \geq 0$ für bestimmte x nach dem Horner-Algorithmus $p_i := p_{i-1}x + a_{n-i}$ für $i = 0, \ldots, n$ mit $p_{-1} = 0$ liefert $p(x) = p_n$. Wir wollen den Horner-Algorithmus dazu benutzen, eine in Binärdarstellung ohne führende Nullen gegebene natürliche Zahl $y = \sum_{i=0}^{n} b_i 2^i$ mit $b_i \in \{0, 1\}$ in eine gleichwertige unäre Strichdarstellung $u(y) \in \{|\}^*$ zu transformieren.

1. Sei $\beta_k = b_{k-1}b_{k-2} \ldots b_0$. Die Berechnung nach Horner protokollieren wir als Folge von „Konfigurationen", d. h. Tripeln $(u(p_i), q, \beta_{n-i})$ mit

   $$u(p_i) = u(2p_{i-1}+b_{n-i}) = u(p_{i-1})u(p_{i-1})u(b_{n-i})\,,$$

   wobei wir $q = q_0 = 0$ setzen wollen.

   Bestimmen Sie $u(9)$, indem Sie die Folge der genannten Konfigurationen für $y = 9$ berechnen.

2. Die Ausführung des obigen Algorithmus stützt sich u. a. auf die Operation der Verdopplung von Strichzahlen. Geben Sie eine (deterministische) Turingmaschine an, die eine eingegebene Strichzahl verdoppelt.

Tutoraufgabe 2

Sei L die zu dem regulären Ausdruck $0|(1(0|1)^*)$ gehörige Sprache. Für eine Zahl $n \in \mathbb{N}$ bezeichne $b(n) \in L$ bzw. $u(n) \in \{|\}^*$ deren binäre bzw. unäre Darstellung.
Es ist eine (deterministische) Ein-Band-Turing-Maschine zu konstruieren, die für $n \neq 0$ die Eingabe $b(n)$ in $u(n)$ umwandelt. Die Eingabe und die Ausgabe sollen sich auf dem sonst leeren Band befinden, und der Lese-/Schreibkopf der Maschine bei Start und Ende der Berechnung soll links auf dem ersten beschriebenen Feld des Bandes stehen.

1. Entwerfen Sie eine Programmstruktur für die zu konstruierende Turingmaschine, indem Sie versuchen, die Gesamtlösung auf Teilprobleme zurückzuführen, für deren Lösung dann in der nachfolgenden Teilaufgabe jeweils spezielle Turingmaschinen implementiert werden können.

2. Implementieren Sie Ihre spezifizierten Turingmaschinen und fügen Sie die Programme zu einem Gesamtprogramm zusammen zur Umwandlung einer binären in eine unäre Darstellung einer natürlichen Zahl.

Tutoraufgabe 3

Wir betrachten den Beweis zu Satz 4.13, in dem eine k-Band-Turingmaschine M durch eine normale Turingmaschine M' simuliert wird. Geben Sie eine Abschätzung für die Anzahl der Zustände an, die M' haben muss.