Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr
Dr. Werner Meixner

Sommersemester 2007
Übungsblatt 8
13. Juni 2007

Einführung in die Theoretische Informatik

Abgabetermin: 18. Juni 2007 **vor** *der Vorlesung oder in den Tutorübungen bis 22. Juni*

Hausaufgabe 1 (5 Punkte)

Beweisen Sie, dass die folgenden Sprachen L_1, L_2 kontextsensitiv sind:

1. $L_1 = \{a^i b^j a^k\,;\, i,j,k \in \mathbb{N}_0,\, j = \max(i,k)\}$,
2. $L_2 = \{a^i b^j a^k\,;\; i,j,k \in \mathbb{N}_0,\, i < j < k\}$.

Hausaufgabe 2 (5 Punkte)

Für Zwecke dieser Aufgabe nennen wir einen deterministischen Kellerautomaten $K = (Q, \Sigma, \Delta, \delta, q_0, Z_0, F)$, dessen Übergangsfunktion δ so beschaffen ist, dass der Kellerinhalt nie verändert wird, einen ϵ-DFA. Sei $L(K)$ die Sprache, die von einem ϵ-DFA K mit Endzuständen akzeptiert wird.

Geben Sie ein direktes (nicht über ϵ-NFA) Verfahren an, das zu einem beliebigen ϵ-DFA K einen deterministischen endlichen Automaten A definiert, der die Sprache $L(K)$ erkennt!

Hausaufgabe 3 (5 Punkte)

Sei $K = (\{p,q\}, \{0,1\}, \{Z_0, X\}, \delta, q, Z_0, \{p\})$ ein Kellerautomat mit folgender Übergangsfunktion

$$\begin{array}{rclrcl}
\delta(q,0,Z_0) & = & \{(q, XZ_0)\}\,, & \delta(q,0,X) & = & \{(q, XX)\}\,, \\
\delta(q,1,X) & = & \{(q, X)\}\,, & \delta(q,\epsilon,X) & = & \{(p,\epsilon)\}\,, \\
\delta(p,\epsilon,X) & = & \{(p,\epsilon)\}\,, & \delta(p,1,X) & = & \{(p, XX)\}\,, \\
\delta(p,1,Z_0) & = & \{(p,\epsilon)\}\,. & & &
\end{array}$$

Berechnen Sie, ausgehend von der Anfangskonfiguration (q, w, Z_0), alle mit folgenden Eingaben erreichbaren Konfigurationen:

i) 01, ii) 0011.

Hausaufgabe 4 (5 Punkte)

Zeigen Sie durch Konstruktion geeigneter linear beschränkter Automaten, dass die Klasse der kontextsensitiven Sprachen abgeschlossen ist gegenüber Durchschnittsbildung.

***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

Es wird erzählt, dass in einem kleinen Dorf in Bayern ein alter Barbier lebt, der alle diejenigen Männer im Dorf rasiert, die sich nicht selbst rasieren.
Warum ist die Frage nach dem Alter des Barbiers sinnlos?
Formalisieren Sie den Sachverhalt und weisen Sie nach, dass die Erzählung eine Lüge enthält!

Vorbereitung 2

Die Ackermann-Funktion $a : \mathbb{N}_0^2 \to \mathbb{N}_0$ ist definiert durch

$$a(x,y) := \begin{cases} y+1\,, & \text{falls } x = 0\,, \\ a(x-1,1)\,, & \text{falls } x \geq 1, y = 0\,, \\ a(x-1, a(x,y-1))\,, & \text{falls } x, y \geq 1\,. \end{cases}$$

Beweisen Sie folgende Eigenschaften der Ackermann-Funktion:

1. $\forall y \in \mathbb{N}_0$: $\quad a(1,y) = y+2,\ a(2,y) = 2y+3\,,$
2. $\forall x, y \in \mathbb{N}_0$: $\quad y < a(x,y)\,.$

Tutoraufgabe 1

Sei $G = (V, \Sigma, P, S)$ eine kontextfreie Grammatik in Chomsky-Normalform ohne nutzlose Variablen mit n Produktionen. Zeigen Sie, dass die Anzahl der Produktionen bei der Überführung in Greibach-Normalform (gemäß Beweis von Satz 77 der Vorlesung) in Abhängigkeit von n exponentiell wachsen kann.

Tutoraufgabe 2

Sei $\Sigma = \{a, b, *\}$, $\Gamma = \Sigma \cup \{\Box\}$ und $Q = \{q_0, q_1, q_2, q_f\}$. Wir betrachten die Turingmaschine $N = (Q, \Sigma, \Gamma, \delta, q_0, \Box, \{q_f\})$ mit der Übergangsfunktion

$$\begin{array}{rclcrcl}
\delta(q_0, a) & = & \{(q_0, a, R)\}\,, & & \delta(q_0, b) & = & \{(q_0, b, R)\}\,, \\
\delta(q_0, *) & = & \{(q_0, a, R)\}\,, & & \delta(q_0, *) & = & \{(q_0, b, R)\}\,, \\
\delta(q_0, \Box) & = & \{(q_1, \Box, L)\}\,, & & & & \\
\delta(q_1, a) & = & \{(q_2, \Box, L)\}\,, & & \delta(q_2, b) & = & \{(q_1, \Box, L)\}\,, \\
\delta(q_1, \Box) & = & \{(q_f, \Box, N)\}\,. & & & &
\end{array}$$

1. Geben Sie eine deterministische Turingmaschine M an, die die Sprache $L(N)$ erkennt.

2. Beschreiben Sie ein allgemeines Verfahren, das zu jeder beliebigen nichtdeterministischen Turingmaschine N eine äquivalente deterministische Turingmaschine M liefert, d. h., so dass $L(N) = L(M)$ gilt.