Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr
Moritz Maaß und Hanjo Täubig

Sommersemester 2005
Übungsblatt 8
10.06.2005

Informatik IV

Abgabetermin: 17.06.2005 **vor** *der Vorlesung*

Aufgabe 1 (10 Punkte)

Ein 2-Kellerautomat $(Q, \Sigma, \Delta, \delta, q_0, Z_0, Z'_0)$ ist ein Kellerautomat, der über einen zweiten Keller verfügt (der mit Z'_0 initialisiert wird). Die Übergangsfunktion

$$\delta : Q \times (\Sigma \cup \{\epsilon\}) \times \Delta \times \Delta \to \mathcal{P}_e(Q \times \Delta^* \times \Delta^*)$$

beschreibt die Vorgehensweise des 2-KA wie folgt ($\mathcal{P}_e$ bezeichnet die Menge aller endlichen Teilmengen). Liest der 2-KA im Zustand q die Eingabe a (auch $a = \epsilon$ möglich), sind Z_1, Z_2 die obersten Zeichen der beiden Keller und gilt $(q', \alpha_1, \alpha_2) \in \delta(q, a, Z_1, Z_2)$, dann kann der 2-KA in den Zustand q' übergehen und hierbei Z_1 durch α_1 und Z_2 durch α_2 ersetzen

(a) Zeigen Sie: Jede 1-Band-Turingmaschine kann durch einen 2-Kellerautomat simuliert werden.

(b) Gilt dies auch, falls der 2-Kellerautomat deterministisch sein soll? (Der 2-KA ist deterministisch falls – in Analogie zu den normalen Kellerautomaten – gilt: $|\delta(q, a, Z_1, Z_2)| + |\delta(q, \epsilon, Z_1, Z_2)| \leq 1$ für alle $(q, a, Z_1, Z_2) \in Q \times \Sigma \times \Delta \times \Delta$.)

Aufgabe 2 (10 Punkte)

Geben Sie eine Turing-Maschine an, die für $n > 0$, falls das Band mit $\text{bin}(n)\#$ initialisiert wird, die Funktion $f(n) = \lfloor \log_2 n \rfloor$ berechnet (das Band soll nach Terminierung mit $\text{bin}(\lfloor \log_2 n \rfloor)\#$ beginnen).

Aufgabe 3 (10 Punkte)

In der Vorlesung wurden WHILE-Programme nur mit den Anweisungen

- $x_i := c$ (Zuweisung)
- $x_i := x_j + c$ (Addition)
- $x_i := x_j \dot{-} c$ (Bedingte Subtraktion)
- $P_1; P_2$ (Sequenz)
- **while** $x_i \neq 0$ **do** P **end** (Schleife)

definiert. Zeigen Sie, dass man die folgenden Anweisungen leicht mittels der oben gegebenen simulieren kann:

- $x_i := x_j + x_k$ (Addition zweier Variablen)

- $x_i := x_j \dot{-} x_k$ (Bedingte Subtraktion zweier Variablen)
- $x_i := x_j * x_k$ (Multiplikation zweier Variablen)
- $x_i := x_j / x_k$ (Ganzzahlige Division zweier Variablen: $x_i = \lfloor \frac{x_j}{x_k} \rfloor$)
- **if** $x_i \neq 0$ **then** P_1 **else** P_2 **fi** (If-Then-Else)

Zeigen Sie ferner, dass man auch Bedingungen der Art $x_i < x_j$, $x_i = x_j$ in den If- und While-Konstrukten simulieren kann.

Aufgabe 4 (10 Punkte)

Geben Sie ein WHILE-Programm an, das den größten gemeinsamen Teiler von x_1 und x_2 berechnet ($x_1, x_2 \in \mathbb{N}_0, x_1, x_2 \neq 0$).

Aufgabe 5 (10 Punkte)

Geben Sie basierend auf den in der Vorlesung definierten fünf Regeln für primitiv rekursive Funktionen jeweils Definitionen für

- $\mathit{add}(n, m) = n + m$
- $\mathit{sub}(n, m) = n \dot{-} m$ (Bedingte Subtraktion, d.h., $\mathit{sub}(n, m) = 0$ für $m \geq n$)
- $\mathit{mul}(n, m) = nm$
- $\mathit{fak}(n) = n!$
- $\mathit{twopow}(n) = 2^n$
- $\mathit{tower}(n) = 2^{2^{2^{\cdot^{\cdot^{\cdot^{2}}}}}}$ (d.h. $2^{(2^{(2^{\cdot^{\cdot^{\cdot^{2}}}})})}$, Turm der Höhe n)
- $\mathit{ifthen}(n, a, b)$ mit

$$\mathit{ifthen}(n, a, b) = \begin{cases} a & n \neq 0 \\ b & n = 0 \end{cases}$$

an.