Klausur Informatik III

2. Oktober 2004

Max Mustermann

1234567

- **Klausur erst nach Aufforderung aufschlagen!**
- Mobiltelefone bitte ausschalten!
- Die Bearbeitungszeit beträgt 60 Minuten.
- Insgesamt sind 60 Punkte erreichbar; die Klausur gilt mit 20 Punkten als bestanden.
- Eine Abgabe ist vor Ablauf der 60 Minuten *nicht* möglich. Bleiben Sie bitte auf Ihrem Platz, bis alle Klausuren eingesammelt wurden.
- Verwenden Sie nur die Blätter des Klausurbogens für Ihre Lösungen. Auf den beiden letzten Blättern finden Sie zusätzlichen Platz.

1	
2	
3	
4	
5	
6	
7	
Σ	
Note	

Max Mustermann
1234567

Aufgabe 1 (16 Punkte)

Es sei $\mathcal{L} = \{a^n b^{n+1} \mid n \geq 0\}$.

a) [3] Geben Sie eine kontextfreie Grammatik Γ über dem Terminalalphabet $\{a, b\}$ an mit $L(\Gamma) = \mathcal{L}$.

b) [6] Zeigen Sie mit Hilfe des Earley-Algorithmus, daß $abb \in L(\Gamma)$.

c) [4] Geben Sie einen Kellerautomaten $\mathcal{A}$ an, der diese Sprache durch den Nullkeller akzeptiert, d.h. $N(\mathcal{A}) = \mathcal{L}$.

d) [3] Zeigen Sie, daß $\mathcal{L}$ nicht regulär ist.

Max Mustermann
1234567

Aufgabe 2 (8 Punkte)

Gegeben sei die Grammatik $\Gamma = \langle\{A, D, K, L, S\}, \{a, b, c, \neg, \vee, \wedge\}, \Pi, S\rangle$, wobei Π folgende Produktionen umfaßt:

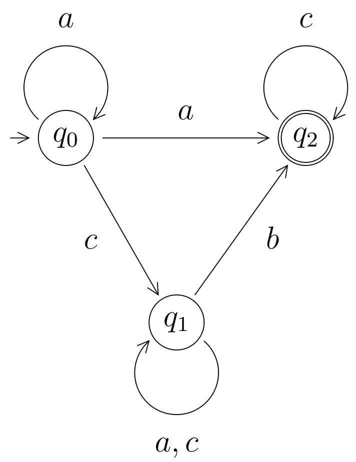
$$\begin{array}{lcl}
S & \longrightarrow & D \\
D & \longrightarrow & L \mid L \vee D \\
K & \longrightarrow & L \mid L \wedge K \\
L & \longrightarrow & A \mid \neg A \\
A & \longrightarrow & a \mid b \mid c
\end{array}$$

Konstruieren Sie eine zu Γ äquivalente Grammatik in Chomsky-Normalform.

Max Mustermann
1234567

Aufgabe 3 (10 Punkte)

Geben Sie zu dem abgebildeten NDEA $\mathcal{A}$ einen regulären Ausdruck γ mit $\langle\gamma\rangle = L(\mathcal{A})$ an. Stellen Sie dazu entweder das zu $\mathcal{A}$ gehörende Gleichungssystem auf und lösen Sie dieses nach q_0, oder verwenden Sie die Kleene-Myhill Konstruktion der $R^k_{i,j}$-Mengen.



Max Mustermann
1234567

Aufgabe 4 (8 Punkte)

Geben Sie Maschinenschemata an, die folgende Funktionen berechnen:

a) [4] die totale Vorgängerfunktion

$$V(x) := \begin{cases} x-1 & \text{falls } x > 0 \\ 0 & \text{falls } x = 0 \end{cases}$$

b) [4] die partielle Vorgängerfunktion

$$V_p(x) := \begin{cases} x-1 & \text{falls } x > 0 \\ \text{undefiniert} & \text{falls } x = 0 \end{cases}$$

Beachten Sie dabei diese beiden Spezifikationen:

- Eine Berechnung führt eine Anfangskonfiguration ($\underline{\#}w$) in eine Haltekonfiguration ($\underline{\#}v$) über.
- Eine natürliche Zahl n werde *unär* codiert, d.h. durch $n+1$ aufeinanderfolgende Markierungen $\underbrace{*\cdots*}_{n+1}$ auf dem Band dargestellt.

Hinweis: Sie können bei *einer* der Teilaufgaben eine Turingmaschine anstatt eines Maschinenschemas angeben.

Max Mustermann
1234567

Aufgabe 5 (4 Punkte)

Geben Sie eine partiell rekursive Definition für folgende Funktion an:

$$x \dot{-} y := \begin{cases} x - y & \text{falls } x \geq y \\ \text{undefiniert} & \text{sonst} \end{cases}$$

Sie dürfen dazu verwenden, daß $+$ eine partiell rekursive Funktion und $=$ ein partiell rekursives Prädikat ist.

Max Mustermann
1234567

Aufgabe 6 (4 Punkte)

Welche der folgenden Behauptungen über Mengen ist richtig?

a) [2] A ist genau dann primitiv rekursiv, wenn sowohl A als auch $\overline{A}$ primitiv rekursiv aufzählbar sind.

b) [2] A ist genau dann rekursiv, wenn sowohl A als auch $\overline{A}$ rekursiv aufzählbar sind.

Begründen Sie Ihre Antworten unter Verweis auf Sätze der Vorlesung.

Max Mustermann
1234567

Aufgabe 7 (10 Punkte)

Gegeben sei das folgende LOOP_2-Programm P:

$$x_2 := 1;\ \textbf{loop}\ x_1\ \textbf{do}\ x_2 := 0\ \textbf{end};\ x_1 := x_2$$

a) [2] Welche aus der Vorlesung bekannte Funktion berechnet P?

b) [8] Es sei β_1 die Belegung $\{x_1 := 2\}$ und β_2 die Belegung $\{x_1 := 0\}$. Geben Sie jeweils die Konfigurationsfolge an, welche die Anfangskonfiguration

$$\begin{pmatrix} \beta_i \\ P; \square \end{pmatrix}$$

in eine Endkonfiguration

$$\begin{pmatrix} \beta_i' \\ \square \end{pmatrix}$$

überführt. Verwenden Sie dazu die folgenden erweiterten Übergangsregeln, *nicht* diejenigen aus der Vorlesung:

- $\begin{pmatrix} \beta \\ x_i := n; P \end{pmatrix} \Longrightarrow \begin{pmatrix} \beta[x_i := n] \\ P \end{pmatrix}$ für jede natürliche Zahl n
- $\begin{pmatrix} \beta \\ x_i := x_j; P \end{pmatrix} \Longrightarrow \begin{pmatrix} \beta[x_i := \beta(x_j)] \\ P \end{pmatrix}$
- $\begin{pmatrix} \beta \\ \textbf{loop}\ x_i\ \textbf{do}\ Q\ \textbf{end}; P \end{pmatrix} \Longrightarrow \begin{pmatrix} \beta \\ \underbrace{Q; \ldots; Q}_{\beta(x_i)-\text{mal}}; P \end{pmatrix}$