Name: [ ] Vorname: [ ] Matr.–Nr.: [ ]

Technische Universität München
Fakultät für Informatik
Prof. Dr. A. Knoll

WS 2002/2003
3. April 2003

Klausur zu Einführung in die Informatik I

Hinweis: Das Aufgabenblatt muss abgegeben werden !

Aufgabe 1 Markow-Algorithmen ((1+2+2+2) + 3 = 10 Punkte)

a) Gesucht ist ein Textersetzungssystem, das natürliche Zahlen $n \geq 0$ in Binärdarstellung. d.h. Wörter der Form $b_k \dots b_1$ mit $b_i \in \{0, L\}$, $1 \leq i \leq k$ invertiert. Beispiel:
das Wort LLL00L wird zu 000LL0 .

(i) Geben Sie einen geeigneten minimalen Zeichenvorrat für das Textersetzungssystem an.

(ii) Geben Sie eine Regelmenge für das Textersetzungssystem und beschreiben Sie kurz Ihre Lösungsidee.

(iii) Testen Sie (unabhängig von der Korrektheit Ihrer Lösung) Ihr Textersetzungssystem mit dem Eingabewort L00LLL00. Geben Sie für jeden Berechnungsschritt die Nummer der angewendeten Regel, und das daraus resultierende Wort an(z.B. L0L0L0L0 $\Rightarrow$ 00L0L0L0 Regel 1, ...).

(iv) Welche der folgende Begriffe treffen für das Textersetzungssytem zu: terminierend, deterministisch, determiniert, sequentiell, parallel.

b) Geben Sie einen Markow-Algorithmus an, der für Wörter über dem Zeichenvorrat $\{a, b, c\}$ und der transitiven Hülle der Relation $B := \{a < b, b < c\}$ eine sortierte Ausgabe gemäß der Relation B liefert. Beispiel:
Eingabewort = bcaaa, Ausgabewort = aaabc

Aufgabe 2 Ringpuffer in OCaml (1+2+2+2+3=10 Punkte)

Ein Ringpuffer stellt eine (First-In First-Out) Warteschlange für maximal n Elemente mit folgenden Eigenschaften dar:

- Werden mehr als (n)-Elemente in den Ring eingetragen, so wird die *älteste* Information überschrieben und geht somit verloren.
- Ein Marker (Schreibzeiger) gebe an, welche der Speicherzellen die neueste Information erhalten soll.
- Ein Marker (Lesezeiger) gebe an, welche der Speicherzellen die älteste Information des Rings beinhaltet.

a) Erstellen Sie eine polymorphe OCaml-Typdeklaration *ringbuffer* gemäß obiger Beschreibung.

b) Erstellen Sie eine OCaml-Funktion *emptyRing*, welche einen Ring mit einem initialen (keine Informationen enthaltenden) Ur-Zustand zurückgibt.

c) Erstellen Sie eine Funktion *memory*, welche die Anzahl der in einem Ringpuffer freien Speicherstellen zurückgibt.

d) Erstellen Sie eine OCaml-Funktion *push*, welches ein neues Element in einen gegebenen Ringpuffer einfügt.

e) Erstellen Sie eine OCaml-Funktion *pop*, bei dessen Aufruf das älteste Element aus einem gegebenen Ringpuffer entfernt, und zusammen mit dem Ringpuffer zurückgegeben wird.

Aufgabe 3 Listen in Prolog (2 + 4 + 4 = 10 Punkte)

a) Eine Liste `A` ganzer Zahlen heißt sortiert,

- wenn die Liste entweder aus höchstens einem Element besteht,
- oder wenn das erste Element kleiner oder gleich dem zweiten Element ist und die Liste ohne das erste Element sortiert ist.

Übertragen Sie diesen Algorithmus in ein Prolog-Programm!

b) Eine Liste `A` enthalte ein Element `E`. Geben Sie ein Prolog-Programm an, das aus der Liste `A` das Element `E` streicht.

c) Eine Liste `B` ist eine Permutation einer Liste `A`, wenn `B` dieselben Elemente wie `A` in einer eventuell anderen Reihenfolge enthält. Geben Sie unter Verwendung des in Teilaufgabe b) implementierten Programms ein Prolog-Programm an, das die Permutationen einer Liste `A` berechnet.

d) (**+2 Punkte**) Definieren Sie unter Verwendung der bisher implementierten Fakten und Regeln ein Prädikat `sortiere`, das eine Liste `A` sortiert.

Aufgabe 4 Termersetzung und strukturelle Induktion (3+7=10 Punkte)

Gegeben sei ein Termersetzungssystem mit folgender rekursiver Definition: ε ist ein Term, und für alle Terme A gilt: $zero(A)$ und $one(A)$ sind auch Terme.

$$
\begin{aligned}
xor(zero(A), zero(B)) &\mapsto zero(xor(A,B)) && (1)\\
xor(one(A), zero(B)) &\mapsto one(xor(A,B)) && (2)\\
xor(zero(A), one(B)) &\mapsto one(xor(A,B)) && (3)\\
xor(one(A), one(B)) &\mapsto zero(xor(A,B)) && (4)\\
xor(\varepsilon, A) &\mapsto A && (5)\\
xor(A, \varepsilon) &\mapsto A && (6)\\
zero(\varepsilon) &\mapsto \varepsilon && (7)
\end{aligned}
$$

a) Vereinfachen Sie folgenden Term mit den angegebenen Regeln so weit wie möglich, d.h. bis zu seiner terminalen Grundform:

$$xor(one(one(one(\varepsilon))), xor(one(one(\varepsilon)), one(zero(one(\varepsilon)))))$$

b) Zeigen Sie (induktiv) daß in dem gegebenen Termersetzungsystem für einen beliebigen Grundterm A immer gilt: Die terminale Grundform von $xor(A,A)$ ist ε.

c) **(+ 3 Punkte)** Sei weiter folgende Interpretation des Termerseztungsystems gegeben:

$$\begin{aligned} zero(A) &\;\hat{=}\; (2 \cdot a) \\ one(A) &\;\hat{=}\; (2 \cdot a) + 1 \\ \varepsilon &\;\hat{=}\; 0 \\ xor(A,B) &\;\hat{=}\; a \oplus b \end{aligned}$$

Zeigen Sie hiermit daß für ein geradzahliges $g \in \mathbb{N}$ und ein ungeradzahliges $u \in \mathbb{N}$ gilt, daß $g \oplus u$ wieder ungeradzahlig ist. (Hinweis: Induktion ist hier unnötig!)