Informatik III WS2004/2005

Übungsblatt 2

Institut Intelligente Systeme
Leiter: Prof. Dr. E. Lehmann
Dietmar Lippold

Besprechungstermin: 01.11. – 05.11.2003

`WWW: http://www.informatik.uni-stuttgart.de/ifi/is/Lehre/Vorlesung/info3.html`

Aufgabe 1

Definieren Sie Prozeduren zu den logischen Funktionen `not`, `or` und `and` nur durch Verwendung der Prozedur `if` und der Boolschen Werte `#t` und `#f`.
Ist bei `or` und `and` eine verkürzte Auswertung möglich, d.h. daß das zweite Argument nicht ausgewertet wird, wenn der Wert der Prozeduren schon durch das erste Argument bestimmt ist?

Aufgabe 2

Beschreiben Sie das Verfahren, nach dem Ausdrücke in Scheme ausgewertet werden. Beschreiben Sie die spezielle Auswertung für Ausdrücke mit `if`, `lambda` und `let` und begründen sie diese. Lesen Sie dazu im R^{5}RS nach (zu finden über die WWW-Seite zur Veranstaltung, s.o.).

Aufgabe 3

In Scheme sind Prozeduren vollwertige Datenobjekte (first order data objects) und dürfen so als Argumente und Werte anderer Prozeduren auftreten. Prozeduren, die andere Prozeduren als Argument oder Wert haben, nennt man *Prozeduren höherer Ordnung*.
Werten Sie folgende Ausdrücke aus:

```
> +          > (define var1 +)          > var1        > (var1 4 5)
> (lambda (x) (* x x))                  > ((lambda (x)(* x x)) 10)
> (define (quad x)(* x x))              > (quad 10)
> (define var2 (lambda (x) (* x x)))    > (var2 10)
> (let ((a  1) (b -2)) (+ 3 (if (< a b) a b)))
> (let ((a  1) (b -2)) ((if (> b 0) + -) a b))
```

Aufgabe 4

Auf welche Weise läßt sich das Ergebnis der Division des Wertes Drei durch den Wert Zwei in Scheme darstellen und worin liegen die Unterschiede?

Aufgabe 5 (schriftlich)

Die Lösungen folgender Aufgabe bitte bis zur Übung dem Tutor der Übungsgruppe, an der Sie teilnehmen, per Email zusenden. Die Email-Adressen stehen auf der WWW-Seite der Lehrveranstaltung.

Ein traditionelles Anwendungsgebiet von Simulation durch mathematische Modelle ist in der Biologie die Entwicklung der Populationsgrößen in einem Ökosystem. Das einfachste Modell berücksichtigt nur eine Art von Lebewesen mit einer Vermehrungsrate g (growth) und einer durch die verfügbare Nahrung oder den beschränkten Lebensraum vorgegebenen festen Maximalpopulation.
Sei p die Population auf [0..1] normierte Population, d.h. Population/Maximalpopulation. Dann wird die neue Population nach einer Generation mit folgender Formel vorhergesagt: $p := g * p * (1 - p)$. Der Term $g * p$ steht für die Vermehrung, der Term $(1 - p)$ berücksichtigt den Tod von Individuen.

1. Implementieren Sie eine Prozedur `(popsim)`. Sie soll Startpopulation, Maximalpopulation, Vermehrungsrate und die gewünschte Anzahl der Generationen vom Benutzer abfragen und anschliessend die Entwicklung der Population simulieren und ausgeben.

   Hinweise: Lesen der Benutzereingabe mit `(read)`. Verwenden Sie für Fragen Strings, z.B. `(display "Wachstumsrate ? ")`.

2. Betrachten Sie Ihre Lösung. Welche Werte bleiben beim Programmlauf unverändert, welche werden bei jeder Generation geändert?

3. Testen Sie die Prozedur mit verschiedenen Parametern. Welche Möglichkeiten gibt es für das Verhalten der Population?

Aufgabe 6

Geben Sie die Ebenen an, auf denen sich die Funktionalität einer Anwendung beschreiben läßt.

Aufgabe 7

Welchen Anforderungen sollte ein Programm genügen, d.h. welche Qualitätsmerkmale gibt es für Programme? Lesen Sie gegebenfalls in Büchern zum Software-Engineering nach.