Mathematisches Institut
der Universität München
Prof. Dr. Helmut Schwichtenberg
Dr. Christian Urban

Wintersemester 2004/2005
Blatt 5

Übungen zum Schemekurs

Aufgabe 16. Man schreibe eine (rekursive) Funktion `flatten`, die einer Liste aus geschachtelten Listen diejenige Liste zuordnet, die aus Nicht-Listen-Elementen aller dieser Listen besteht.
Beispiel: `(flatten '((a b) ((c) d (e f g)))) => (a b c d e f g)`.

Aufgabe 17. Die *Church Numerale* werden erzeugt durch

```
(define zero (lambda (f) (lambda (x) x)))

(define (succ n)
  (lambda (f) (lambda (x) (f ((n f) x)))))
```

Man definiere `one` (d.h., `(succ zero)`) und `two` (d.h., `(succ (succ zero))`) direkt—also nicht aus `zero` und `succ`. Ferner gebe man eine direkte Definition der Additionsprozedur `add` (also nicht durch wiederholte Anwendung von `succ`).

Aufgabe 18. Man betrachte das folgende Programm

```
(define ff
   (let ((state #f))
     (lambda ()
       (set! state (not state))
       state)))
```

und erkläre was passiert, wenn man die folgenden Ausdrücke evaluiert.

```
(ff)
(ff)
(ff)
```

Als nächstes betrachte man das folgende Programm

```
(define (c start)
  (let ((state start))
    (lambda ()
      (set! state (+ state 1))
      state)))
```

und erkläre was passiert, wenn man die folgenden Ausdrücke evaluiert.

```
(define c1 (c 0))
(define c2 (c 0))
(c1)
(c2)
(c2)
(c2)
(c1)
```

Abgabe. Montag, den 11. Oktober 2004, vor der Vorlesung oder (besser) per Email an urban@mathematik.uni-muenchen.de