Mathematisches Institut
der Universität München
H. Schwichtenberg
L. Crosilla, M. Seisenberger

Wintersemester 2002/2003
Blatt 4

Übungen zum Ferienkurs über Nichtnumerisches Programmieren

Aufgabe 13. Ist f eine einstellige Funktion und n eine natürliche Zahl, so ist die n-fache Anwendung von f definiert als die Abbildung, die jedem Argument x zuordnet $f(\dots f(f(x))\dots)$. Man schreibe eine Prozedur `repeated`, die aus einem einstelligen f und einer Zahl n die n-fache Anwendung von f konstruiert. Was ist der Wert von `((repeated square 2) 5)`?

Aufgabe 14. (a) Man schreibe eine einstellige Prozedur `reverse`, die einer Liste die umgedrehte Liste zuordnet.

(b) Man schreibe eine einstellige Prozedur `last-pair`, die einer Liste das letzte in ihr verwendete Paar zuordnet (dessen `cdr`-Feld dann einen Zeiger auf die leere Liste enthält).

Aufgabe 15. Die Prozedur `square-list` soll die Elemente einer Liste `items` quadrieren. Man definiere sie auf zwei verschiedene Arten, durch Ergänzung der folgenden Muster.

```
(define (square-list items)
  (if (null? items)
      '()
      (cons ?? ??)))

(define (square-list items)
  (map ?? ??))
```

Aufgabe 16. Man implementiere `union-set` für die Darstellung von Mengen als geordnete und ungeordnete Listen.

Aufgabe 17. Sei F_n die n-te Fibonacci-Zahl.

(a) Man zeige, daß mit $A = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 1 \end{smallmatrix}\right) \in \mathbb{N}^{2\times 2}$ gilt

$$A^n \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} F_n \\ F_{n+1} \end{pmatrix}.$$

(b) Man verwende dies zum Schreiben einer Prozedur `fib` zur Berechnung der Fibonacci-Zahlen mit Zeitkomplexität $O(\log(n))$.

Abgabetermin. Montag, den 7. Oktober 2002, in der Übung oder (besser) per email an `seisenb@mathematik.uni-muenchen.de`