Christian-Albrechts-Universität zu Kiel
Institut für Informatik
Lehrstuhl für Programmiersprachen und Übersetzerkonstruktion
Prof. Dr. M. Hanus, F. Reck

Institut für Informatik

13. Übung zur Vorlesung „Informatik I“

Abgabe am Freitag, 6. Februar, 14 Uhr

Hinweis: Die Anmeldung zur Nachschreibklausur am 30.3.2009 wird im Zeitraum vom 9.3.2009 bis zum 22.3.2009 über die StudiDB möglich sein.

Präsenzaufgabe 1

Definieren Sie eine Prozedur `(prune! n l)`, die eine positive Zahl `n` und eine Liste `l`, die mindestens die Länge `n` hat, als Parameter erwartet. Als Seiteneffekt soll die übergebene Liste auf die angegebene Länge gekürzt werden. Zum Beispiel soll nach den folgenden Eingaben

```
(define xs (list 1 2 3))
(prune! 2 xs)
```

die Variable `xs` den Wert `(list 1 2)` haben. Beschreiben Sie die Auswertung der angegeben Eingaben im Umgebungsmodell.

Aufgabe 2 — 6 Punkte

Für eine ganzzahlige Folge $f : \mathbb{N} \to \mathbb{Z}$ ist die *diskrete Ableitung* definiert als:

$$f'(x) = f(x+1) - f(x)$$

Höhere Ableitungen werden rekursiv definiert:

$$f^{(0)} = f, \; f^{(n+1)} = {f^{(n)}}'$$

Implementieren Sie eine Scheme-Funktion `(disk-ableitung n s)`, die die `n`-te diskrete Ableitung der als unendlicher Strom `s` dargestellten Folge f

$(f(0)\ \ f(1)\ \ f(2)\ \ \ldots)$

berechnet.

Aufgabe 3 — 6 Punkte

Wir wollen in dieser Aufgabe Vektoren $v = (v_i)$ als Datenströme von Zahlen darstellen und Matrizen $m = (m_{ij})$ als Datenströme von Vektoren (die Zeilen der Matrix). Zum Beispiel soll die Matrix:

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \end{pmatrix}$$

durch den Strom *((1 2 3 4) (5 6 7 8) (9 10 11 12))* dargestellt werden. In dieser Darstellung sollen nun die grundlegenden Operationen für Matrizen und Vektoren mit Hilfe der Operationen für Datenströme ausgedrückt werden. Implementieren Sie die folgenden Matrix-Operationen:

- `(skalar-produkt v w)`, liefert für zwei Vektoren `v` = $(v_1 \ldots v_n)$ und `w` = $(w_1 \ldots w_n)$ das Ergebnis $\sum_{i=1}^{n} v_i w_i$.

- `(matrix-vektor-mult m v)`, liefert für eine $k \times n$ - Matrix `m` = $\begin{pmatrix} m_{11} & \ldots & m_{1n} \\ \ldots & \ldots & \ldots \\ m_{k1} & \ldots & m_{kn} \end{pmatrix}$ und einen Vektor `v` = $(v_1 \ldots v_n)$ den Vektor $(t_1 \ldots t_k)$ mit $t_i = \sum_{j=1}^{n} m_{ij} v_j$.

- `(matrix-mult m m')`, liefert für eine $k \times l$ - Matrix `m` = $\begin{pmatrix} m_{11} & \ldots & m_{1l} \\ \ldots & \ldots & \ldots \\ m_{k1} & \ldots & m_{kl} \end{pmatrix}$ und eine $l \times n$ - Matrix `m'` = $\begin{pmatrix} m'_{11} & \ldots & m'_{1n} \\ \ldots & \ldots & \ldots \\ m'_{l1} & \ldots & m'_{ln} \end{pmatrix}$ die $k \times n$ - Matrix $\begin{pmatrix} m''_{11} & \ldots & m''_{1n} \\ \ldots & \ldots & \ldots \\ m''_{k1} & \ldots & m''_{kn} \end{pmatrix}$ mit $m''_{ij} := \sum_{h=1}^{l} m_{ih} m'_{hj}$.

- `(transponiere m)`, liefert für eine Matrix `m` = $\begin{pmatrix} m_{11} & \ldots & m_{1n} \\ \ldots & \ldots & \ldots \\ m_{k1} & \ldots & m_{kn} \end{pmatrix}$ die Matrix $\begin{pmatrix} m_{11} & \ldots & m_{k1} \\ \ldots & \ldots & \ldots \\ m_{1n} & \ldots & m_{kn} \end{pmatrix}$.

Aufgabe 4 2+2+2+2 Punkte

Implementieren Sie in Java die folgenden Klassenmethoden:

- `int fakRek(int n)` zur rekursiven Berechnung der Fakultätsfunktion,
- `int fakIter(int n)` zur iterativen Berechnung dieser mit einer Schleife,
- `int fibRek(int n)` zur rekursiven Berechnung der Fibonacci Funktion und
- `int fibIter(int n)` zur iterativen Berechnung dieser mit einer Schleife.

Über die Übungswebsite wird ihnen ein Gerüst `FakFib.java` zur Verfügung gestellt, das die Methodenköpfe bereits enthält. Das Gerüst enthält auch eine `main` Methode zum Testen Ihrer Implementierung. Sie können es mit dem Kommando

```
javac FakFib.java
```

übersetzen und die übersetzte Klasse mit

```
java FakFib
```

ausführen. Bitte fügen Sie Ihrer Abgabe die dabei erzeugte Ausgabe bei.