Übungsaufgaben zum SML-Kurs, WS 2000/2001

Hans Leiß

Abgabe: Di, 9.1.2001

4 Funktoren zur Bildung von Matrizenringen

Aufgabe 4.1 (i) Definiere zu geeigneten Argument- und Ergebnissignaturen `SIG` und `SIG'` einen Funktor

⟨*nsqMatrix*⟩≡

```
functor nsqMatrix (X : SIG) : SIG' = struct ... end
```

mit dem man aus einer Eingabe `X:SIG`, in der ein Halbring `R:HALBRING` und eine Zahl `n:int` verpackt sind, den (Halb-) Ring `nsqMatrix(X)` aller $n \times n$- Matrizen über R berechnen kann, falls $n \geq 0$ ist. (6 Punkte)

(ii) Bilde mit Hilfe von `nsqMatrix` den Matrizenring aller 4×4-Matrizen über dem Ring `IntRing` der ganzen Zahlen (vgl. Vorlesungsfolien) und berechne die Matrizen $B = A + A$ und $C = A \cdot A$ für

$$A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 4 & 3 & 2 & 1 \\ -0 & -1 & -2 & -3 \\ -3 & -2 & -1 & -0 \end{pmatrix}$$

(4 Punkte)

Aufgabe 4.2 Implementiere den Funktor

⟨*Matrix*⟩≡

```
functor Matrix (R : HALBRING) : MATRIX = struct ... end
```

anders als in der Vorlesung: eine Matrix der Dimension $n \times m$ soll jetzt durch ein Tripel `(f, n, m)` dargestellt werden, wobei `f` eine Funktion `f:int * int -> R.t` ist, n die Zeilen- und m die Spaltenzahl der Matrix. (8 Punkte)

Bemerkung 4.1 Aufgabe 3.3 enthielt einen Fehler. Man kann `dnf` nur im Halbring `Prop` definieren, nicht in einem beliebigen Halbring, da man dafür mit *Ausdrücken* für Ringelemente rechnet.

Schöne Weihnachtsferien!
H.Leiß