

Informatik II, SoSe 2008

Übungsblatt 10

Abgabe: bis Mi 02.07.2008 7:30 Uhr

Besprechung 03.07.08 - 08.07.08

**Aufgabe 10-1** schriftlich bearbeiten/SML  
**Tiefendurchlauf**

Die Datei 10-1.sml enthält die Definitionen des Datentyps 'a binbaum5 und der Funktion depth\_first\_collect aus der Vorlesung.

- a) Beschreiben Sie im vorbereiteten SML-Kommentar in der Datei, was folgende Funktion leistet:

```
fun test(x,B) =  
  depth_first_collect true  
    (fn m      => x=m)  
    (fn (m1,m2) => x=m2 orelse x=m1)  
    (fn (R1,m,R2) => x=m orelse (R1 andalso R2))  
  B;
```

Die Beschreibung soll nicht den Ablauf des Programms in natürliche Sprache übersetzen („wenn test auf einen Baum angewandt wird, dessen Wurzel mit Knt3 gebildet ist, dann wird test zunächst rekursiv auf die beiden Teilbäume angewandt...“), sondern erklären, welcher Zusammenhang zwischen Element x und Baum B getestet wird („test(x,B) hat den Wert true genau dann wenn...“)

- b) **datatype** farbe = rot | orange | gold | gelb | gruen | blau | violett  
**datatype** deco = kerze **of** bool (\* kerze true: brennende Kerze \*)  
              | kugel **of** farbe (\* kugel rot : rote Kugel \*)  
              | lametta **of** int (\* lametta m : m Lamettafaeden \*)  
              | stern **of** int (\* stern n : mit Leuchtkraft n \*)

Ein Baum mit Knotenmarkierungen vom Typ deco heißt ein Weihnachtsbaum, wenn auf jedem seiner Äste eine brennende Kerze vorkommt.

Schreiben Sie eine Funktion ist\_weihnachtsbaum vom Typ deco binbaum5 -> bool, die entscheidet, ob ihr Argument ein Weihnachtsbaum ist. Die Datei enthält einen Beispielsbaum w vom Typ deco binbaum5, der *kein* Weihnachtsbaum ist. Es reicht aber, eine einzige weitere Kerze anzuzünden, damit er ein Weihnachtsbaum ist – wenn es die richtige Kerze ist.

**Aufgabe 10-2** Muster vs. if-then-else

Bartl Bastscho braucht eine Funktion, die ihm sagt, wie lang die Jahreszeiten auf der Nordhalbkugel sind. Er überlegt, ob er sie mit der Muster-Technik oder mit der if-then-else-Technik formulieren soll:

```
datatype jahreszeit = fruehling | sommer | herbst | winter;
```

```
fun laenge fruehling = 93.0  
  | laenge sommer   = 93.0  
  | laenge herbst    = 89.5  
  | laenge winter    = 89.5
```

```
fun laenge j =  
  if      j=fruehling then 93.0  
  else if j=sommer   then 93.0  
  else if j=herbst    then 89.5  
  else (* j=winter *)    89.5
```

Er fragt Vroni Frogstmi, was ihr besser gefällt. Unvorsichtigerweise lässt er dabei die Bemerkung fallen, dass die Entscheidung zwischen Muster und if-then-else wohl nur von rein stilistischen Gesichts-

punkten abhängt. Vroni erwidert, dann würde sie gern mal sehen, wie er die folgenden Funktionen mit beiden Techniken definieren kann:

- Für den Datentyp `int` sei `betrag` die Funktion, die jede positive ganze Zahl auf sich selbst abbildet und jede nicht-positive auf das  $(-1)$ -fache.
- Für **`datatype`** `koordinaten = koord of int * int` seien `x_wert` und `y_wert` die Funktionen, die die jeweilige Koordinate liefern.

Untersuchen Sie, welche dieser Funktionen mit welcher der beiden Techniken definiert werden kann. Leiten Sie daraus allgemeinere Antworten zu folgenden Fragen ab:

- In welchen Fällen ist eine Definition mit `if-then-else` möglich, aber nicht mit Mustern?
- In welchen Fällen ist eine Definition mit Mustern möglich, aber nicht mit `if-then-else`?

### Aufgabe 10-3      schriftlich bearbeiten Typisierbarkeit

Geben Sie für jede der folgenden Funktionen entweder den Typ mit einer Begründung an, wie er zustandekommt, oder eine Begründung, warum die Funktion keinen Typ hat. Die Begründungen sollen nicht in einem Formalismus gegeben werden, sondern in natürlicher Sprache.

- a) `fun a(x) = if x>=0 then x else ~1.0`
- b) `fun b(x,y) = if x then y+1 else x`
- c) `fun c(x) = c(x=x)`
- d) `fun d(x) = if true then d(x) else d(x)=d(x)`