TECHNISCHE UNIVERSITÄT MÜNCHEN
FAKULTÄT FÜR INFORMATIK
Lehrstuhl für Sprachen und Beschreibungsstrukturen WS 2015/16
Einführung in die Informatik 2 **Übungsblatt 11**
Prof. Dr. Helmut Seidl, Ralf Vogler, Stefan Schulze Frielinghaus

Aufgabe 11.1 [3+3 Punkte] In großen Schritten zum Ziel

Gegeben seien folgende MiniOCaml-Definitionen:

```
let f = fun x -> x+1*2
let g = fun x -> x+42
let h = 3
```

Konstruieren Sie Beweise für folgende Aussagen:

a) $\texttt{match}\ [1;2;3]\ \texttt{with}\ [\,]\ \texttt{->}\ 0 \mid x :: xs\ \texttt{->}\ f\ x \Rightarrow 3$

b) $\texttt{let}\ x = 5\ \texttt{in}\ g\ x + h \Rightarrow 50$

Lösungsvorschlag 11.1

Um die Beweisbäume möglichst lesbar zu halten, wurde auf die Benutzung von $v \Rightarrow v$ verzichtet.

a)

$$\text{PM}\ \dfrac{[1;2;3] \Rightarrow 1 :: [2;3] \qquad \text{App}\ \dfrac{\text{GD}\ \dfrac{f = \texttt{fun}\ x\ \texttt{->}\ x+1*2}{f \Rightarrow \texttt{fun}\ x\ \texttt{->}\ x+1*2} \qquad \text{Op}\ \dfrac{\text{Op}\ \dfrac{1*2 \Rightarrow 2}{1*2 \Rightarrow 2} \qquad 1+2 \Rightarrow 3}{1+1*2 \Rightarrow 3}}{f\ 1 \Rightarrow 3}}{\texttt{match}\ [1;2;3]\ \texttt{with}\ [\,]\ \texttt{->}\ 0 \mid x :: xs\ \texttt{->}\ f\ x \Rightarrow 3}$$

b)

$$\text{LD}\ \dfrac{\text{Op}\ \dfrac{\text{App}\ \dfrac{\text{GDef}\ \dfrac{g = \texttt{fun}\ x\ \texttt{->}\ x+42}{g \Rightarrow \texttt{fun}\ x\ \texttt{->}\ x+42} \qquad \text{Op}\ \dfrac{5+42 \Rightarrow 47}{5+42 \Rightarrow 47}}{g\ 5 \Rightarrow 47} \qquad \text{GD}\ \dfrac{h = 3}{h \Rightarrow 3} \qquad 47+3 \Rightarrow 50}{g\ 5 + h \Rightarrow 50}}{\texttt{let}\ x = 5\ \texttt{in}\ g\ x + h \Rightarrow 50}$$

Aufgabe 11.2 [4+3 Punkte] Terminierung

Gegeben sei nun:

```
let rec doit a x = if x = 0 then a else doit (a+2*x-1) (x-1)
let square x = doit 0 x
```

a) Zeigen Sie, dass `square` für alle positiven `x` terminiert.

b) (Bonus) Zeigen Sie, dass `square` nicht immer terminiert.

Lösungsvorschlag 11.2

Angepasst auf MiniOCaml:

```
let rec doit = fun a -> fun x -> match x with 0 -> a | x -> doit (a+2*x-1) (x-1)
let square = fun x -> doit 0 x
```

Wir nehmen folgende Setzungen vor:

$$e_{match} = \texttt{match}\ x\ \texttt{with}\ 0\ \texttt{->}\ a\ \mid\ x\ \texttt{->}\ \texttt{doit}\ (a+2*x-1)(x-1)$$

$$e_{funx} = \texttt{fun x ->}\ e_{match}$$

$$\pi_a = \textsc{App}\;\dfrac{\text{GD}\;\dfrac{\texttt{doit = fun a ->}\ e_{funx}}{\texttt{doit} \Rightarrow \texttt{fun a ->}\ e_{funx}} \quad e_{funx}[a/a] \Rightarrow e_{funx}[a/a]}{\texttt{doit a} \Rightarrow e_{funx}[a/a]}$$

a) Induktionsanfang $x = 0$:

$$\textsc{App}\;\dfrac{\pi_a \quad \text{PM}\;\dfrac{0 \Rightarrow 0 \qquad a[a/a] \Rightarrow a}{e_{match}[0/x] \Rightarrow a}}{\texttt{(doit a) 0} \Rightarrow a}$$

Induktionsschritt $x > 0$:

$$\textsc{App}\;\dfrac{\pi_a \quad \text{PM}\;\dfrac{\text{Op}\;\dfrac{0+2*1-1 \Rightarrow 1 \qquad \texttt{(doit 1) 0} \Rightarrow 1}{1 \Rightarrow 1 \qquad \texttt{(doit (0+2*x-1) (x-1))}[1/x] \Rightarrow 1}}{e_{match}[0/a, 1/x] \Rightarrow 1}}{\texttt{(doit 0) 1}}$$

b) Wir nehmen an, dass `square` auch für negative Zahlen terminiert. Um zu terminieren müssten wir den Basisfall `(doit a) 0` erreichen. Dies ist jedoch nicht möglich:

$$\textsc{App}\;\dfrac{\pi_a \quad \text{PM}\;\dfrac{\text{Op}\;\dfrac{0+2*(-1)-1 \Rightarrow -3 \qquad \textsc{App}\;\dfrac{\dots}{\texttt{(doit (-3)) (-2)} \Rightarrow ?}}{-1 \Rightarrow -1 \qquad \texttt{(doit (0+2*x-1) (x-1))}[-1/x] \Rightarrow ?}}{e_{match}[0/a, -1/x] \Rightarrow ?}}{\texttt{(doit 0) (-1)}}$$

Aufgabe 11.3 Tutoraufgaben

Gegeben seien folgende MiniOCaml-Definitionen:

```
let f = fun x -> (fun y -> 2 * x + y)

let g = f 7

let rec fact =
  fun n ->
    match n with
      0 -> 1
    | x -> x * (fact (x-1))
```

Konstruieren Sie Beweise für folgende Aussagen:

a) `f 3 4` $\Rightarrow$ `10`

b) `g` $2 \Rightarrow 16$

c) `fact` $2 \Rightarrow 2$

Lösungsvorschlag 11.3
Um die Beweisbäume möglichst lesbar zu halten, wurde auf die Benutzung von $v \Rightarrow v$ verzichtet.

a)

$$\textsc{App}\ \dfrac{\textsc{GD}\ \dfrac{f = \texttt{fun}\ x\ \texttt{->}\ (\texttt{fun}\ y\ \texttt{->}\ 2*x+y)}{f \Rightarrow \texttt{fun}\ x\ \texttt{->}\ (\texttt{fun}\ y\ \texttt{->}\ 2*x+y)} \qquad \textsc{Op}\ \dfrac{\textsc{Op}\ \dfrac{2*3 \Rightarrow 6}{2*3 \Rightarrow 6} \qquad 6+4 \Rightarrow 10}{2*3+4 \Rightarrow 10}}{f\ 3\ 4 \Rightarrow 10}$$

b)

$$\textsc{App}\ \dfrac{\textsc{GD}\ \dfrac{g = f\ 7 \qquad \textsc{App}\ \dfrac{\textsc{GD}\ \dfrac{f = \texttt{fun}\ x\ \texttt{->}\ (\texttt{fun}\ y\ \texttt{->}\ 2*x+y)}{f \Rightarrow \texttt{fun}\ x\ \texttt{->}\ (\texttt{fun}\ y\ \texttt{->}\ 2*x+y)}}{f\ 7 \Rightarrow \texttt{fun}\ y\ \texttt{->}\ 2*7+y}}{g \Rightarrow \texttt{fun}\ y\ \texttt{->}\ 2*7+y} \qquad \textsc{Op}\ \dfrac{\textsc{Op}\ \dfrac{2*7 \Rightarrow 14}{2*7 \Rightarrow 14} \qquad 14+2 \Rightarrow 16}{2*7+2 \Rightarrow 16}}{g\ 2 \Rightarrow 16}$$

c) Zur Vereinfachung nehmen wir folgende Setzungen vor:

$$e_{match} = \texttt{match}\ n\ \texttt{with}\ 0\ \texttt{->}\ 1\ \mid\ x\ \texttt{->}\ x*(\texttt{fact}\ (x-1))$$

$$\pi =$$

$$\textsc{GD}\ \dfrac{\texttt{fact} = \texttt{fun}\ n\ \texttt{->}\ e_{match}}{\texttt{fact} \Rightarrow \texttt{fun}\ n\ \texttt{->}\ e_{match}}$$

$$
\textsc{App}\ \pi\ \dfrac{
\textsc{PM}\ \dfrac{
2 \Rightarrow 2 \equiv x[2/x]
\qquad
\textsc{Op}\ \dfrac{
\textsc{App}\ \pi\ \dfrac{
\textsc{Op}\ \dfrac{2 - 1 \Rightarrow 1}{2 - 1 \Rightarrow 1}
\qquad
\textsc{PM}\ \dfrac{
1 \Rightarrow 1 \equiv x[1/x]
\qquad
\textsc{Op}\ \dfrac{
\textsc{App}\ \pi\ \dfrac{
\textsc{Op}\ \dfrac{1 - 1 \Rightarrow 0}{1 - 1 \Rightarrow 0}
\qquad
\textsc{PM}\ \dfrac{0 \Rightarrow 0 \equiv 0}{e_{match}[0/n] \Rightarrow 1}
}{\texttt{fact}\ (1 - 1) \Rightarrow 1}
\qquad
\textsc{Op}\ \dfrac{1 * 1 \Rightarrow 1}{1 * 1 \Rightarrow 1}
}{1 * (\texttt{fact}\ (1 - 1)) \Rightarrow 1}
}{e_{match}[1/n] \Rightarrow 1}
}{\texttt{fact}\ (2 - 1) \Rightarrow 1}
\qquad
2 * 1 \Rightarrow 2
}{2 * (\texttt{fact}\ (2 - 1)) \Rightarrow 2}
}{e_{match}[2/n] \Rightarrow 2}
}{\texttt{fact}\ 2}
$$