TECHNISCHE UNIVERSITÄT MÜNCHEN FAKULTÄT FÜR INFORMATIK



Übungsblatt 11

Lehrstuhl für Sprachen und Beschreibungsstrukturen Einführung in die Informatik 2

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 = \text{fun } x \rightarrow x+1*2
let g = \text{fun } x \rightarrow x+42
let h = 3
```

Konstruieren Sie Beweise für folgende Aussagen:

```
a) match [1;2;3] with [\ ] -> 0\mid x::xs -> f\mid x\Rightarrow 3
```

b) let x = 5 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.

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