Functional Programming and Verification

Prof. Dr. H. Seidl, N. Hartmann, R. Vogler

WS 2018/19

Exercise Sheet 11

Deadline: 20.01.2019

General Information

Detailed information about the lecture, tutorials and homework assignments can be found on the lecture website[1]. Solutions have to be submitted to Moodle[2]. Make sure your uploaded documents are readable. Blurred images will be rejected. Use Piazza[3] to ask questions and discuss with your fellow students.

Big-step proofs

Unless specified otherwise, all rules used in a big-step proof tree must be annotated and all axioms ($v \Rightarrow v$) must be written down.

Assignment 11.1 (L) Big Steps

We define these functions:

```
let rec f = fun l ->
  match l with [] -> 1 | x::xs -> x + g xs
and g = fun l ->
  match l with [] -> 0 | x::xs -> x * f xs
```

Consider the following expressions. Find the values they evaluate to and construct a big-step proof for that claim.

1. `let f = fun a -> (a+1,a-1)::[] in f 7`
2. `f [3;6]`
3. `(fun x -> x 3) (fun y z -> z y) (fun w -> w + w)`

Suggested Solution 11.1

1. Big step tree:

$$\pi_0 = \text{LI}\ \dfrac{\text{TU}\ \dfrac{\text{OP}\ \dfrac{7 \Rightarrow 7 \quad 1 \Rightarrow 1 \quad 7+1 \Rightarrow 8}{7+1 \Rightarrow 8} \quad \text{OP}\ \dfrac{7 \Rightarrow 7 \quad 1 \Rightarrow 1 \quad 7-1 \Rightarrow 6}{7-1 \Rightarrow 6}}{(7+1,7-1) \Rightarrow (8,6)}}{[(7+1,7-1)] \Rightarrow [(8,6)]}$$

[1] https://www.in.tum.de/i02/lehre/wintersemester-1819/vorlesungen/functional-programming-and-verification/

[2] https://www.moodle.tum.de/course/view.php?id=44932

[3] https://piazza.com/tum.de/fall2018/in0003/home

$$\text{LD}\ \frac{\texttt{fun a -> [(a+1,a-1)]} \Rightarrow \texttt{fun a -> [(a+1,a-1)]} \quad \text{APP'}\ \frac{\texttt{fun a -> [(a+1,a-1)]} \Rightarrow \texttt{fun a -> [(a+1,a-1)]}\ \texttt{7} \Rightarrow \texttt{7}\ \ \pi_0}{\texttt{(fun a -> [(a+1,a-1)]) 7} \Rightarrow \texttt{[(8,6)]}}}{\texttt{let f = fun a -> [(a+1,a-1)] in f 7} \Rightarrow \texttt{[(8,6)]}}$$

2. Big step tree:

$$\pi_f = \text{GD}\ \frac{\texttt{f} = \texttt{fun l -> match l with [] -> 1 | x::xs -> x+g xs}\ \ \texttt{fun l -> match l with [] -> 1 | x::xs -> x+g xs} \Rightarrow \texttt{fun l -> match l with [] -> 1 | x::xs -> x+g xs}}{\texttt{f} \Rightarrow \texttt{fun l -> match l with [] -> 1 | x::xs -> x+g xs}}$$

$$\pi_g = \text{GD}\ \frac{\texttt{g} = \texttt{fun l -> match l with [] -> 0 | x::xs -> x*f xs}\ \ \texttt{fun l -> match l with [] -> 0 | x::xs -> x*f xs} \Rightarrow \texttt{fun l -> match l with [] -> 0 | x::xs -> x*f xs}}{\texttt{g} \Rightarrow \texttt{fun l -> match l with [] -> 0 | x::xs -> x*f xs}}$$

$$\pi_0 = \text{PM}\ \frac{[6] \Rightarrow [6] \quad \text{OP}\ \frac{6 \Rightarrow 6 \quad \text{APP'}\ \frac{\pi_f\ \ [] \Rightarrow [] \quad \text{PM}\ \frac{[] \Rightarrow []\ \ 1 \Rightarrow 1}{\texttt{match [] with [] -> 1 | x::xs -> x+g xs} \Rightarrow 1}}{\texttt{f [] } \Rightarrow 1} \quad 6*1 \Rightarrow 6}{\texttt{6*f [] } \Rightarrow 6}}{\texttt{match [6] with [] -> 0 | x::xs -> x*f xs} \Rightarrow 6}$$

$$\text{APP'}\ \frac{\pi_f\ \ [3;6] \Rightarrow [3;6] \quad \text{PM}\ \frac{[3;6] \Rightarrow [3;6] \quad \text{OP}\ \frac{3 \Rightarrow 3 \quad \text{APP'}\ \frac{\pi_g\ \ [6] \Rightarrow [6]\ \ \pi_0}{\texttt{g } [6] \Rightarrow 6} \quad 3+6 \Rightarrow 9}{\texttt{3+g } [6] \Rightarrow 9}}{\texttt{match } [3;6] \texttt{ with [] -> 1 | x::xs -> x+g xs} \Rightarrow 9}}{\texttt{f } [3;6] \Rightarrow 9}$$

3. Big step tree:

$$\pi_0 = \text{APP'}\ \frac{\texttt{fun x -> x 3} \Rightarrow \texttt{fun x -> x 3}\ \ \texttt{fun y z -> z y} \Rightarrow \texttt{fun y z -> z y} \quad \text{APP'}\ \frac{\texttt{fun y z -> z y} \Rightarrow \texttt{fun y z -> z y}\ \ \texttt{3} \Rightarrow \texttt{3}\ \ \texttt{fun z -> z 3} \Rightarrow \texttt{fun z -> z 3}}{\texttt{(fun y z -> z y) 3} \Rightarrow \texttt{fun z -> z 3}}}{\texttt{(fun x -> x 3) (fun y z -> z y)} \Rightarrow \texttt{fun z -> z 3}}$$

$$\text{APP'}\ \frac{\pi_0\ \ \texttt{fun w -> w+w} \Rightarrow \texttt{fun w -> w+w} \quad \text{APP'}\ \frac{\texttt{fun w -> w+w} \Rightarrow \texttt{fun w -> w+w}\ \ \texttt{3} \Rightarrow \texttt{3} \quad \text{OP}\ \frac{\texttt{3} \Rightarrow \texttt{3}\ \ \texttt{3} \Rightarrow \texttt{3}\ \ \texttt{3+3} \Rightarrow \texttt{6}}{\texttt{3+3} \Rightarrow \texttt{6}}}{\texttt{(fun w -> w+w) 3} \Rightarrow \texttt{6}}}{\texttt{(fun x -> x 3) (fun y z -> z y) (fun w -> w+w)} \Rightarrow \texttt{6}}$$

Assignment 11.2 (L) Multiplication

Prove that the function

```
let rec mul a b =
  match a with 0 -> 0 | _ -> b + mul (a-1) b
```

terminates for all inputs $a, b \geq 0$.

Suggested Solution 11.2

We prove by induction on `a` that `mul a b` terminates with $a * b$:

- Base case: `a = 0`:

$$\text{APP}\ \frac{\pi_{mul} \quad \text{PM}\ \frac{}{\texttt{match 0 with 0 -> 0 | _ -> b + mul (-1) b} \Rightarrow \texttt{0}}}{\texttt{mul 0 b} \Rightarrow \texttt{0}}$$

- Inductive case: Assume `mul a b` terminates for an $a \geq 0$. Now, we show that it also terminates for $a + 1$:

$$\text{APP}\ \frac{\pi_{mul}\quad \text{PM}\ \dfrac{\text{OP}\ \dfrac{\text{APP}\ \dfrac{\text{by I.H.}}{\texttt{mul (a+1-1) b} \Rightarrow a*b}\quad b + (a*b) \Rightarrow (a+1)*b}{\texttt{b + mul (a+1-1) b} \Rightarrow (a+1)*b}}{\texttt{match a+1 with 0 -> 0 | \_ -> b + mul (a+1-1) b} \Rightarrow (a+1)*b}}{\texttt{mul (a+1) b} \Rightarrow (a+1)*b}$$

Here π_{mul} is the GD-tree of `mul`. Note one important thing here: When reducing to the induction hypothesis, we do not apply the operator rule for the `a+1-1` term, since $a+1$ is not really an OCaml expression, but the successor of a. We silently simplify $a+1-1$ to a and apply the induction hypothesis.

□

Assignment 11.3 (L) Threesum

Use big-step operational semantics to show that the function

```
let rec threesum = fun l ->
  match l with [] -> 0 | x::xs -> 3*x + threesum xs
```

terminates for all inputs and computes three times the sum of the input list's elements.

Suggested Solution 11.3

We define:

$$\pi_{ts} = \text{GD}\ \frac{\texttt{threesum = fun l -> match l with [] -> 0 | x::xs -> 3*x + threesum xs}}{\texttt{threesum} \Rightarrow \texttt{fun l -> match l with [] -> 0 | x::xs -> 3*x + threesum xs}}$$

Now, we do an induction on the length n of the list.

- Base case: $n = 0$ (`l = []`)

$$\text{APP}\ \frac{\pi_{ts}\ \texttt{[]} \Rightarrow \texttt{[]}\quad \text{PM}\ \dfrac{\texttt{[]} \Rightarrow \texttt{[]}\quad \texttt{0} \Rightarrow \texttt{0}}{\texttt{match [] with [] -> 0 | x::xs -> 3*x + threesum xs} \Rightarrow \texttt{0}}}{\texttt{threesum []} \Rightarrow \texttt{0}}$$

- Inductive step: We assume `threesum xs` terminates with $3\sum_{i=1}^{n} x_i$ for an input `xs` $= [x_n; \ldots; x_1]$ of length $n \geq 0$. Now, show that `threesum` x_{n+1}`::xs` terminates with $3\sum_{i=1}^{n+1} x_i$:

$$\text{APP}\ \frac{\pi_{ts}\ x_{n+1}\texttt{::xs} \Rightarrow x_{n+1}\texttt{::xs}\quad \text{PM}\ \dfrac{x_{n+1}\texttt{::xs} \Rightarrow x_{n+1}\texttt{::xs}\quad \text{OP}\ \dfrac{\texttt{3} \Rightarrow \texttt{3}\ \ x_{n+1} \Rightarrow x_{n+1}\ \ \texttt{3}*x_{n+1} \Rightarrow 3x_{n+1}}{3*x_{n+1} \Rightarrow 3x_{n+1}}\quad \text{APP}\ \dfrac{\text{by I.H.}}{\texttt{threesum xs} \Rightarrow 3\sum_{i=1}^{n} x_i}\quad 3x_{n+1}+3\sum_{i=1}^{n} x_i \Rightarrow 3\sum_{i=1}^{n+1} x_i}{3*x_{n+1}\ \texttt{+ threesum xs} \Rightarrow 3\sum_{i=1}^{n+1} x_i}}{\texttt{match}\ x_{n+1}\texttt{::xs with [] -> 0 | x::xs -> 3*x + threesum xs} \Rightarrow 3\sum_{i=1}^{n+1} x_i}}{\texttt{threesum (}x_{n+1}\texttt{::xs)} \Rightarrow 3\sum_{i=1}^{n+1} x_i}$$

□

Assignment 11.4 (L) Records

Let MiniOCaml++ be an extended version of MiniOCaml that comes with records. Perform these tasks:

1. Extend the operational big-step semantics of MiniOCaml for these new expressions.
2. Construct a big-step proof for the value of this expression:

```
let r = { x={ a=3+5; b=2+4::[] }; y=2*7 } in r.x.a::r.x.b
```

Suggested Solution 11.4

1. We need two new rules for the record evaluation (RE) and record access (RA):

   - RE $\dfrac{e_1 \Rightarrow v_1 \quad \ldots \quad e_n \Rightarrow v_n}{\{a_1 = e_1;\ \ldots\ a_n = e_n\} \Rightarrow \{a_1 = v_1;\ \ldots\ a_n = v_n\}}$
   - RA $\dfrac{e \Rightarrow \{\ \ldots\ a = v;\ \ldots\ \}}{e.a \Rightarrow v}$

2. The big-step tree:

$$\pi_0 = \text{RE}\ \frac{\text{RE}\ \dfrac{\text{OP}\ \dfrac{3 \Rightarrow 3 \quad 5 \Rightarrow 5 \quad 3+5 \Rightarrow 8}{3+5 \Rightarrow 8}\ \text{LI}\ \dfrac{\text{OP}\ \dfrac{2 \Rightarrow 2 \quad 4 \Rightarrow 4 \quad 2+4 \Rightarrow 6}{2+4 \Rightarrow 6}}{[2+4] \Rightarrow [6]}}{\{\ \texttt{a=3+5;\ b=[2+4]}\ \} \Rightarrow \{\ \texttt{a=8;\ b=[6]}\ \}}\ \text{OP}\ \dfrac{2 \Rightarrow 2 \quad 7 \Rightarrow 7 \quad 2*7 \Rightarrow 14}{2*7 \Rightarrow 14}}{\{\ \texttt{x=\{ a=3+5; b=[2+4] \}; y=2*7}\ \} \Rightarrow \{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \}}$$

$$\pi_1 = \text{RA}\ \frac{\text{RA}\ \dfrac{\{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \} \Rightarrow \{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \}}{\{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \} \Rightarrow \{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \}}}{\{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \}\texttt{.x} \Rightarrow \{\ \texttt{a=8; b=[6]}\ \}}$$

$$\pi_2 = \text{RA}\ \frac{\text{RA}\ \dfrac{\{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \} \Rightarrow \{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \}}{\{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \} \Rightarrow \{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \}}}{\{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \}\texttt{.x} \Rightarrow \{\ \texttt{a=8; b=[6]}\ \}}$$

$$\text{LD}\ \frac{\pi_0 \quad \text{LI}\ \dfrac{\pi_1 \quad \pi_2}{\{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \}\texttt{.x.a::}\{\ \texttt{x=\{ a=8; b=[6] \}; y=14}\ \}\texttt{.x.b} \Rightarrow [8;6]}}{\texttt{let r = \{ x=\{ a=3+5; b=[2+4] \}; y=2*7 \} in r.x.a::r.x.b} \Rightarrow [8;6]}$$

Assignment 11.5 (H) More Big Steps [12 Points]

Globally defined are these functions:

```
let rec map = fun f l ->
  match l with [] -> [] | x::xs -> f x :: map f xs
and fold_left = fun f a l ->
  match l with [] -> a | x::xs -> fold_left f (f a x) xs
and comp = fun f g x -> f (g x)
and mul = fun a b -> a * b
and id = fun x -> x
```

Give big-step proofs for the following claims:

1. `fold_left mul 3 [10]` $\Rightarrow$ `30`
2. `(let a = comp (fun x -> 2 * x) in a (fun x -> x + 3)) 4` $\Rightarrow$ `14`
3. `map (id id) [8]` $\Rightarrow$ `[8]`

Suggested Solution 11.5

1.

$$\pi_{fl} = \texttt{fun f a l -> match l with [] -> a | x::xs -> fold\_left f (f a x) xs}$$

$$\pi_{fold\_left} = \text{GD}\ \frac{\texttt{fold\_left} = \pi_{fl}\ \ \pi_{fl} \Rightarrow \pi_{fl}}{\texttt{fold\_left} \Rightarrow \texttt{fun f a l -> match l with [] -> a | x::xs -> fold\_left f (f a x) xs}}$$

$$\pi_{mul} = \text{GD}\ \frac{\texttt{mul} = \texttt{fun a b -> a*b}\ \ \texttt{fun a b -> a*b} \Rightarrow \texttt{fun a b -> a*b}}{\texttt{mul} \Rightarrow \texttt{fun a b -> a*b}}$$

$$\pi_0 = \text{APP'}\ \frac{\texttt{fun a b -> a*b} \Rightarrow \texttt{fun a b -> a*b}\ \ 3 \Rightarrow 3 \to 10 \Rightarrow 10 \quad \text{OP}\ \dfrac{3 \Rightarrow 3\ \ 10 \Rightarrow 10\ \ 3*10 \Rightarrow 30}{3*10 \Rightarrow 30}}{\texttt{(fun a b -> a*b) 3 10} \Rightarrow 30}$$

$$\pi_1 = \text{PM}\ \frac{[] \Rightarrow []\ \ 30 \Rightarrow 30}{\texttt{match [] with [] -> 30 | x::xs -> fold\_left (fun a b -> a*b) ((fun a b -> a*b) 30 x) xs} \Rightarrow 30}$$

$$\pi_2 = \text{PM}\ \frac{[10] \Rightarrow [10] \quad \text{APP'}\ \dfrac{\pi_{fold\_left}\ \ \texttt{fun a b -> a*b} \Rightarrow \texttt{fun a b -> a*b}\ \pi_0\ \ [] \Rightarrow []\ \ \pi_1}{\texttt{fold\_left (fun a b -> a*b) ((fun a b -> a*b) 3 10) []} \Rightarrow 30}}{\texttt{match [10] with [] -> 3 | x::xs -> fold\_left (fun a b -> a*b) ((fun a b -> a*b) 3 x) xs} \Rightarrow 30}$$

$$\text{APP'}\ \frac{\pi_{fold\_left}\ \ \pi_{mul}\ \ 3 \Rightarrow 3\ \ [10] \Rightarrow [10]\ \ \pi_2}{\texttt{fold\_left mul 3 [10]} \Rightarrow 30}$$

2.

$$\pi_{comp} = \text{GD}\ \frac{\texttt{comp} = \texttt{fun f g x -> f (g x)}\ \ \texttt{fun f g x -> f (g x)} \Rightarrow \texttt{fun f g x -> f (g x)}}{\texttt{comp} \Rightarrow \texttt{fun f g x -> f (g x)}}$$

$$\pi_{fun} = \texttt{fun x -> (fun x -> 2*x) ((fun x -> x+3) x)} \Rightarrow \texttt{fun x -> (fun x -> 2*x) ((fun x -> x+3) x)}$$

$$\pi_0 = \text{APP'}\ \frac{\pi_{comp}\ \ \texttt{fun x -> 2*x} \Rightarrow \texttt{fun x -> 2*x}\ \ \texttt{fun g x -> (fun x -> 2*x) (g x)} \Rightarrow \texttt{fun g x -> (fun x -> 2*x) (g x)}}{\texttt{comp (fun x -> 2*x)} \Rightarrow \texttt{fun g x -> (fun x -> 2*x) (g x)}}$$

$$\pi_1 = \text{LD}\ \frac{\pi_0 \quad \text{APP'}\ \dfrac{\texttt{fun g x -> (fun x -> 2*x) (g x)} \Rightarrow \texttt{fun g x -> (fun x -> 2*x) (g x)}\ \ \texttt{fun x -> x+3} \Rightarrow \texttt{fun x -> x+3}\ \ \pi_{fun}}{\texttt{(fun g x -> (fun x -> 2*x) (g x)) (fun x -> x+3)} \Rightarrow \texttt{fun x -> (fun x -> 2*x) ((fun x -> x+3) x)}}}{\texttt{let a = comp (fun x -> 2*x) in a (fun x -> x+3)} \Rightarrow \texttt{fun x -> (fun x -> 2*x) ((fun x -> x+3) x)}}$$

$$\text{APP'}\ \frac{\pi_1\ \ 4 \Rightarrow 4 \quad \text{APP'}\ \dfrac{\texttt{fun x -> 2*x} \Rightarrow \texttt{fun x -> 2*x} \quad \text{APP'}\ \dfrac{\texttt{fun x -> x+3} \Rightarrow \texttt{fun x -> x+3}\ \ 4 \Rightarrow 4 \quad \text{OP}\ \dfrac{4 \Rightarrow 4\ \ 3 \Rightarrow 3\ \ 4+3 \Rightarrow 7}{4+3 \Rightarrow 7}}{\texttt{(fun x -> x+3) 4} \Rightarrow 7} \quad \text{OP}\ \dfrac{2 \Rightarrow 2\ \ 7 \Rightarrow 7\ \ 2*7 \Rightarrow 14}{2*7 \Rightarrow 14}}{\texttt{(fun x -> 2*x) ((fun x -> x+3) 4)} \Rightarrow 14}}{\texttt{(let a = comp (fun x -> 2*x) in a (fun x -> x+3)) 4} \Rightarrow 14}$$

3.

$$\pi_{id} = \text{GD}\ \frac{\texttt{id} = \texttt{fun x -> x}\ \ \texttt{fun x -> x} \Rightarrow \texttt{fun x -> x}}{\texttt{id} \Rightarrow \texttt{fun x -> x}}$$

$$\pi_m = \texttt{fun f l -> match l with [] -> [] | x::xs -> f x::map f xs}$$

$$\pi_{map} = \text{GD}\ \frac{\texttt{map} = \pi_m\ \ \pi_m \Rightarrow \pi_m}{\texttt{map} \Rightarrow \texttt{fun f l -> match l with [] -> [] | x::xs -> f x::map f xs}}$$

$$\pi_0 = \text{APP'}\ \frac{\pi_{map}\ \ \texttt{fun x -> x} \Rightarrow \texttt{fun x -> x}[] \Rightarrow [] \quad \text{PM}\ \dfrac{[] \Rightarrow []\ \ [] \Rightarrow []}{\texttt{match [] with [] -> [] | x::xs -> (fun x -> x) x::map (fun x -> x) xs} \Rightarrow []}}{\texttt{map (fun x -> x) []} \Rightarrow []}$$

$$\pi_1 = \text{LI}\ \frac{\text{APP'}\ \dfrac{\texttt{fun x -> x} \Rightarrow \texttt{fun x -> x}\ \ 8 \Rightarrow 8\ \ 8 \Rightarrow 8}{\texttt{(fun x -> x) 8} \Rightarrow 8} \quad \pi_0}{\texttt{(fun x -> x) 8::map (fun x -> x) []} \Rightarrow [8]}$$

$$\text{APP'}\ \frac{\pi_{map} \quad \text{APP'}\ \dfrac{\pi_{id}\ \ \pi_{id}\ \ \texttt{fun x -> x} \Rightarrow \texttt{fun x -> x}}{\texttt{id id} \Rightarrow \texttt{fun x -> x}} \quad [8] \Rightarrow [8] \quad \text{PM}\ \dfrac{[8] \Rightarrow [8]\ \ \pi_1}{\texttt{match [8] with [] -> [] | x::xs -> (fun x -> x) x::map (fun x -> x) xs} \Rightarrow [8]}}{\texttt{map (id id) [8]} \Rightarrow [8]}$$

Assignment 11.6 (H) Computing Zero [4 Points]

Consider the function `foo`:

```
let rec foo = fun l ->
  match l with [] -> 0
  | 0::xs -> foo xs
  | x::xs -> if x > 0 then foo (x-1::xs) else foo (x+1::xs)
```

Prove that `foo` terminates for all inputs. Axioms ($v \Rightarrow v$) may be omitted.

Suggested Solution 11.6

Let $\pi_{foo} =$ $\text{GD}\ \dfrac{\texttt{foo = fun l -> ... fun l -> ...} \Rightarrow \texttt{fun l -> ...}}{\texttt{foo} \Rightarrow \texttt{fun l -> ...}}$

We prove by induction on the length n of list `l`.

- Base case: $n = 0$ (`l = []`):

  $$\text{APP}\ \frac{\pi_{foo}\ \ \text{PM}\ \texttt{match [] with [] -> 0 | ...} \Rightarrow \texttt{0}}{\texttt{foo []} \Rightarrow \texttt{0}}$$

- Inductive step: We assume `foo xs` terminates with 0 for a list `xs` of length $n \geq 0$. Now, we show that it also terminates for a list `x::xs` of length $n+1$. To do so, we do another induction on the value of the first element `x` of the list:

  - Base case: $x = 0$:

    $$\text{APP}\ \frac{\pi_{foo}\ \ \text{PM}\ \dfrac{\text{APP}\ \dfrac{\text{by I.H.}}{\texttt{foo xs} \Rightarrow \texttt{0}}}{\texttt{match 0::xs with ... | 0::xs -> foo xs | ...} \Rightarrow \texttt{0}}}{\texttt{foo (0::xs)} \Rightarrow \texttt{0}}$$

  - Inductive step: We assume `foo x::xs` terminates with 0 for a value $x \geq 0$. Now, we show that `foo (x+1::xs)` also terminates:

    $$\text{APP}\ \frac{\pi_{foo}\ \ \text{PM}\ \dfrac{\text{PM}\ \dfrac{\text{APP}\ \dfrac{\text{by I.H.}}{\texttt{foo (x+1-1::xs)} \Rightarrow \texttt{0}}}{\texttt{match x+1 > 0 with true -> foo (x+1-1::xs) | ...} \Rightarrow \texttt{0}}}{\texttt{match (x+1::xs) with ... | x::xs -> match x > 0 with true -> foo (x-1::xs) | ...} \Rightarrow \texttt{0}}}{\texttt{foo (x+1::xs)} \Rightarrow \texttt{0}}$$

  - Inductive step We assume `foo x::xs` terminates with 0 for a value $x \leq 0$. Now, we show that `foo (x-1::xs)` also terminates:

    $$\text{APP}\ \frac{\pi_{foo}\ \ \text{PM}\ \dfrac{\text{PM}\ \dfrac{\text{APP}\ \dfrac{\text{by I.H.}}{\texttt{foo (x-1+1::xs)} \Rightarrow \texttt{0}}}{\texttt{match x-1 > 0 with ... | false -> foo (x-1+1::xs) | ...} \Rightarrow \texttt{0}}}{\texttt{match x-1::xs with ... | x::xs -> match x > 0 with ... | false -> foo (x+1::xs)} \Rightarrow \texttt{0}}}{\texttt{foo (x-1::xs)} \Rightarrow \texttt{0}}$$

□

Assignment 11.7 (H) Raise the bar! [4 Points]

Given are these definitions:

```
let rec impl = fun n a ->
  match n with 0 -> a | _ -> impl (n-1) (a * n * n)
and bar = fun n -> impl n 1
```

Prove that `bar n` terminates with $n! * n!$ for all non-negative inputs n. Axioms ($v \Rightarrow v$) may be omitted.

Suggested Solution 11.7

We first prove that `impl n a` terminates with $a * n! * n!$: We omit the global definitions of π_{impl} and π_{bar} of `impl` and `bar` here for simplicity.

- Base case: $n = 0$:

$$\text{APP'}\ \frac{\pi_{impl}\quad \text{PM}\ \dfrac{}{\texttt{match 0 with 0 -> a | ...} \Rightarrow a}}{\texttt{impl 0 a} \Rightarrow a}$$

- Inductive step: We assume `impl n a` terminates with $a * n! * n!$ for an input $n \geq 0$. Now, we show that it terminates with $a * (n+1)! * (n+1)!$ for input $n + 1$:

$$\text{APP'}\ \frac{\pi_{impl}\quad \text{PM}\ \dfrac{\text{APP'}\ \dfrac{\text{by I.H.}}{\texttt{impl (n+1-1) (a * (n+1) * (n+1))} \Rightarrow (a * (n+1) * (n+1)) * n! * n!}}{\texttt{match n+1 with ... | \_ -> impl (n+1-1) (a * (n+1) * (n+1))} \Rightarrow (a * (n+1) * (n+1)) * n! * n!}}{\texttt{impl (n+1) a} \Rightarrow a * (n+1)! * (n+1)!}$$

Now, that we have shown that `impl n a` $\Rightarrow a * n! * n!$ (1), we prove:

$$\text{APP}\ \frac{\pi_{bar}\quad \text{APP'}\ \dfrac{\text{by (1)}}{\texttt{impl n 1} \Rightarrow n! * n!}}{\texttt{bar n} \Rightarrow n! * n!}$$

□