ETH zürich

Department of Computer Science

Formal Methods and Functional Programming – Assignment 2

Author: Yannis Huber (17-570-110)

Teaching assistant: Tobias Oberdörfer

Typing proofs and inference

a)

$\lambda x.(x$ `1 True,` x `0) :: (Int -> Bool -> a) -> (a, Bool -> a)`:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma\vdash\texttt{True :: }\tau_1}\,\text{True}\quad \triangle}{\Gamma\vdash x\ \texttt{1 True :: a}}\,\text{App}\qquad \dfrac{\dfrac{}{\Gamma\vdash x\texttt{:: }\tau_0\texttt{ -> Bool -> a}}\,\text{Var}\quad \dfrac{}{\Gamma\vdash 0\texttt{ :: }\tau_0}\,\text{Int}}{\Gamma\vdash x\ \texttt{0 :: Bool -> a}}\,\text{App}}{\Gamma := x\texttt{ : (Int -> Bool -> a)}\vdash(x\ \texttt{1 True, }x\ \texttt{0) :: (a, Bool -> a)}}\,\text{Tuple}}{\vdash\lambda x.(x\ \texttt{1 True, }x\ \texttt{0) :: (Int -> Bool -> a) -> (a, Bool -> a)}}\,\text{Abs}$$

Where $\triangle$ is:

$$\dfrac{\dfrac{}{\Gamma\vdash x\texttt{ :: }\tau_2\texttt{ -> }\tau_1\texttt{ -> a}}\,\text{Var}\quad \dfrac{}{\Gamma\vdash\texttt{1 :: }\tau_2}\,\text{Int}}{\Gamma\vdash x\ \texttt{1 :: }\tau_1\texttt{ -> a}}\,\text{App}$$

Constraints: τ_0 = `Int`, τ_1 = `Bool`, τ_2 = `Int`.

b)

$(\lambda x.\lambda y.(y$ `(iszero (`$y\ x$`)))) True ::` τ_0:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma\vdash y\texttt{ :: }\tau_4\texttt{ -> }\tau_3}\,\text{Var}\qquad \dfrac{\dfrac{\dfrac{}{\Gamma\vdash y\texttt{ :: }\tau_5\texttt{ -> Int}}\,\text{Var}\quad \dfrac{}{\Gamma\vdash x\texttt{ :: }\tau_5}\,\text{Var}}{\Gamma\vdash y\ x\texttt{ :: Int}}}{\Gamma\vdash\texttt{iszero (}y\ x\texttt{) :: }\tau_4}\,\text{iszero}}{\Gamma := x : \tau_1,\ y : \tau_2\vdash y\ \texttt{(iszero (}y\ x\texttt{)) :: }\tau_3}\,\text{App}}{x : \tau_1\vdash\lambda y.(y\ \texttt{(iszero (}y\ x\texttt{))) :: }\tau_0}\,\text{Abs}}{\vdash\lambda x.\lambda y.(y\ \texttt{(iszero (}y\ x\texttt{))) :: }\tau_1\texttt{ -> }\tau_0}\,\text{Abs}\qquad \dfrac{}{\vdash\texttt{True :: }\tau_1}\,\text{True}}{\vdash(\lambda x.\lambda y.(y\ \texttt{(iszero (}y\ x\texttt{)))) True :: }\tau_0}\,\text{App}$$

Constraints: $\tau_2 = \tau_4$ `->` τ_3, τ_4 = `Bool`, $\tau_1 = \tau_5$, $\tau_2 = \tau_5$ `-> Int`, τ_1 = `Bool`, $\tau_0 = \tau_2$ `->` τ_3.
This gives us τ_0 = `(Bool -> Int) -> Int`.

c)

$\lambda x.\lambda y.$ `if` $y\ x$ `then (fst` x`) else (snd (snd` x`)) ::` τ_0:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma\vdash y\texttt{ :: }\tau_5\texttt{ -> Bool}}\,\text{Var}\quad \dfrac{}{\Gamma\vdash x\texttt{ :: }\tau_5}\,\text{Var}}{\Gamma\vdash y\ x\texttt{ :: Bool}}\quad \dfrac{\dfrac{}{\Gamma\vdash x\texttt{ :: (}\tau_4\texttt{, }\tau_6\texttt{)}}\,\text{Var}}{\Gamma\vdash\texttt{fst }x\texttt{ :: }\tau_4}\,\text{fst}\quad \dfrac{\dfrac{\dfrac{}{\Gamma\vdash x\texttt{ :: (}\tau_8\texttt{, (}\tau_7\texttt{,}\tau_4\texttt{))}}\,\text{Var}}{\Gamma\vdash\texttt{snd }x\texttt{ :: (}\tau_7\texttt{,}\tau_4\texttt{)}}}{\Gamma\vdash\texttt{snd (snd }x\texttt{) :: }\tau_4}\,\text{snd}}{\Gamma := x : \tau_1, y : \tau_3\vdash\texttt{if }y\ x\texttt{ then (fst }x\texttt{) else (snd (snd }x\texttt{)) :: }\tau_4}\,\text{if}}{x : \tau_1\vdash\lambda y.\texttt{ if }y\ x\texttt{ then (fst }x\texttt{) else (snd (snd }x\texttt{)) :: }\tau_2}\,\text{Abs}}{\vdash\lambda x.\lambda y.\texttt{ if }y\ x\texttt{ then (fst }x\texttt{) else (snd (snd }x\texttt{)) :: }\tau_0}\,\text{Abs}$$

Constraints: $\tau_0 = \tau_1$ `->` τ_2, $\tau_2 = \tau_3$ `->` τ_4, $\tau_3 = \tau_5$ `-> Bool`, $\tau_1 = \tau_5$, τ_1 = `(`τ_4`, `τ_6`)`, τ_1 = `(`τ_8`, (`τ_7`, `τ_4`))`.
This gives us τ_0 = `(a, (b, c)) -> ((a, (b, c)) -> Bool) -> c`.

d)

This is an example for an invalid expression. This type cannot be constructed because the deduction leads to a unification error between `Int` and `(Int,` τ_1`)`.

$$\dfrac{\dfrac{\dfrac{}{\vdash \texttt{3+5 :: (Int, } \tau_1\texttt{)}}\ \lightning}{\vdash \texttt{fst (3+5) :: Int}}\ \text{fst}}{\vdash \texttt{iszero (fst (3+5)) :: } \tau_0}\ \text{iszero}$$

e)

Here, the derivation also leads to an unification error between `Bool` and τ_2 `->` τ_1. Therefore, the expression is invalid.

$$\dfrac{\dfrac{\dfrac{}{x \texttt{ : } \tau_1 \vdash 0 \texttt{ :: } \tau_0}\ \text{Int}}{\vdash \lambda x.0 \texttt{ :: } \tau_1 \texttt{ -> } \tau_0}\ \text{Abs} \qquad \dfrac{\dfrac{}{\vdash \texttt{True :: } \tau_2 \texttt{ -> } \tau_1}\ \lightning \qquad \dfrac{}{\vdash \texttt{5 :: } \tau_2}\ \text{Int}}{\vdash \texttt{True 5 :: } \tau_1}\ \text{App}}{\vdash (\lambda x.0) \texttt{ (True 5) :: } \tau_0}\ \text{App}$$