HW05

Tuesday, 31. October 2023 09:33

Problems marked with num were corrected

1.

a)

$$\dfrac{\dfrac{}{f: \text{Bool} \to \text{Bool} \vdash f: \text{Bool} \to \text{Bool}}\,Ax \qquad \dfrac{\dfrac{}{\Gamma \vdash \text{false}: \text{Bool}}\,T\text{-false} \quad \dfrac{}{\Gamma' \vdash \text{true}: \text{Bool}}\,T\text{-true} \quad \dfrac{}{\Gamma \vdash \text{false}: \text{Bool}}\,T\text{-false}}{\Gamma \vdash \text{if false then true else false}: \text{Bool}}\,T\text{-If}}{\underbrace{f: \text{Bool} \to \text{Bool}}_{\Gamma} \vdash f\,(\text{if false then true else false}): \text{Bool}}$$

b)

$$\dfrac{\dfrac{\dfrac{}{f: \text{Bool} \to \text{Bool} \vdash f: \text{Bool} \to \text{Bool}}\,Ax \qquad \dfrac{\dfrac{}{x: \text{Bool} \vdash x: \text{Bool}}\,Ax \quad \dfrac{}{x: \text{Bool} \vdash x: \text{Bool}}\,Ax \quad \dfrac{}{\Gamma \vdash \text{false}: \text{Bool}}\,T\text{-false}}{\Gamma \vdash (\text{if } x \text{ then false else } x): \text{Bool}}\,T\text{-If}}{\underbrace{f: \text{Bool} \to \text{Bool},\ x}_{\Gamma}: \text{Bool}.(\text{if } x \text{ then false else } x): \text{Bool} \to \text{Bool}}\,E}{f: \text{Bool} \to \text{Bool} \vdash \lambda x: \text{Bool}.\, f(\text{if } x \text{ then false else } x): \text{Bool} \to \text{Bool}}\,I$$

2.

a) $(\alpha \to \beta) \to (\beta \to \gamma) \to \alpha \to \gamma$

à la Church: $\lambda x^{\alpha \to \beta}.\, \lambda y^{\beta \to \gamma}.\, \lambda z^{\alpha}.\, y\,(x\,z)$

à la Curry: $\lambda x.\, \lambda y.\, \lambda z.\, y\,(x\,z)$

Our term has to take three inputs, since if $\alpha \to \gamma$ would be the output, the term will not be well typed. Therefore this is the only inhabitant.

b) $\alpha \to \beta \to (\alpha \to \beta \to \gamma) \to \gamma$

à la Church: $\lambda x^{\alpha}.\, \lambda y^{\beta}.\, \lambda z^{\alpha \to \beta \to \gamma}.\, z\,x\,y$

à la Curry: $\lambda x.\, \lambda y.\, \lambda z.\, z\,x\,y$

The output type of the term has to be γ and as inputs our term takes two variable types and a function which will be applied to them and γ type would be returned. Therefore no other inhabitants are possible.

c) $((\alpha \to \beta \to \alpha) \to \alpha) \to \alpha$

à la Church: $\lambda x^{((\alpha \to \beta \to \alpha) \to \alpha)}.\, x\,(\lambda f^{\alpha} g^{\beta}.\, f)$

à la Curry: $\lambda x.\, x\,(\lambda y.\, y\,(\lambda f g.\, f))$

This is a more complex case, however it leaves us with only one option for an inhabitant, as clearly our term takes only one argument and has to "internally define" other functions in order for the term to be well-typed.

d) $\beta \to ((\alpha \to \beta) \to \gamma) \to \gamma$

à la Church: $\lambda x^{\beta}.\, \lambda y^{(\alpha \to \beta) \to \gamma}.\, y\,(\lambda z^{\alpha}.\, x)$

à la Curry: $\lambda x.\, \lambda y.\, y\,(\lambda z.\, x)$

Our inner function has to take a type α but return β which we have defined as λx. This allows us to intrinsically define λz to conform with the well-type-edness of the term.

e) $\alpha \to (\alpha \to \alpha) \to \alpha$

à la Church: $\lambda x^{\alpha}.\, \lambda y^{\alpha \to \alpha}.\, y\,x$ or $\lambda x^{\alpha}.\, \lambda y^{\alpha \to \alpha}.\, x$ or $\lambda x^{\alpha}.\, \lambda y^{\alpha \to \alpha}.\, y\,(y\,x)$ ...

à la Curry: $\lambda x.\, \lambda y.\, y\,x$ or $\lambda x.\, \lambda y.\, x$ or $\lambda x. \lambda y.\, y\,(y\,x)$ and so on...

3.

a) $S = \lambda x y z.\, x z\,(y z)$

$\lambda x: \ ? \to \gamma$

$\lambda y: \ ? \to \beta$

$\lambda z: \ \alpha$

$y z \Rightarrow z: \alpha \to \beta \Rightarrow \lambda y: \alpha \to \beta \Rightarrow yz: \beta$

$x\,z\,(y z) \Rightarrow x \ z: \alpha \ \ yz: \beta \Rightarrow x: \alpha \to \beta \to \gamma$

ANSWER:

$(\alpha \to \beta \to \gamma) \to (\alpha \to \beta) \to \alpha \to \gamma$

b) $K = \lambda x y.\, x = \lambda x^{\alpha}.\, \lambda y^{\beta}.\, x : \alpha$

ANSWER:

$\alpha \to \beta \to \alpha$

c) $SKK = (\lambda x y z.\, x z (y z))\,K K =$

$= (\lambda y z.\, K z (y z))\,K =$

$= \lambda z.\, K z\,(K z) =$

$= \lambda z.\, K z\,((\lambda x y.\, x)\, z) =$

$= \lambda z.\, K z z =$

$= \lambda z.\, z = \lambda z^{\alpha}.\, z : \alpha$ (id)

ANSWER: $\alpha \to \alpha$ (Identity)

d) $I = \lambda x.\, x = \lambda x^{\alpha}.\, x : \alpha$ (id)

ANSWER: $\alpha \to \alpha$ (Identity)

4.

$((\alpha \to \beta) \to \gamma) \to \beta \to \gamma$ – type derivation

$\lambda x^{(\alpha \to \beta) \to \gamma}.\, \lambda y^{\beta}.\, x\,(\lambda z^{\alpha}.\, z)\, y$ – corresponding term

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\Gamma' \vdash \gamma}\,Ax}{\Gamma, (\alpha \to \beta) \vdash \gamma}\,I}{\Gamma \vdash (\alpha \to \beta) \to \gamma}\,I \qquad \dfrac{}{\Gamma \vdash \alpha \to \beta}\,Ax}{\Gamma \equiv ((\alpha \to \beta) \to \gamma), \beta \vdash \gamma}\,E}{((\alpha \to \beta) \to \gamma) \vdash \beta \to \gamma}\,I}{\vdash ((\alpha \to \beta) \to \gamma) \to \beta \to \gamma}\,I$$

5.

a)

$M, N = x \mid MN \mid (M, N) \mid \lambda x: T.N \mid \langle N \mid \mid N \rangle$

$T, \sigma = \alpha_i \mid T \to \sigma \mid \sigma \times T$

b)

$$\dfrac{\Gamma \vdash a: \sigma \qquad \Gamma \vdash b: T}{\Gamma \vdash (a, b): \sigma \times T}\,\text{Prod}$$

$$\dfrac{\Gamma \vdash a: \sigma \times T}{\Gamma \vdash \langle a \mid : \sigma}\,\text{Left}$$

$$\dfrac{\Gamma \vdash b: \sigma \times T}{\Gamma \vdash \mid b \rangle : T}\,\text{Right} \quad ???$$

6. X

7.

a) fact c_3 where fact c_n = if (iszero c_n) then c_1 else times c_n (fact (pred c_n))

fact c_3 = if iszero c_3 then c_1 else times c_3 (fact (pred c_3)) =

= if iszero c_3 then c_1 else times c_3 (fact c_2) =

= if iszero c_3 then c_1 else times c_3 (if iszero c_2 then c_1 else times c_2 (fact (pred c_2))) =

= if iszero c_3 then c_1 else times c_3

(if iszero c_2 then c_1 else times c_2 (fact c_1)) =

= if iszero c_3 then c_1 else times c_3 (if iszero c_2 then c_1 else times c_2 ->

-> (if iszero c_1 then c_1 else times c_1 (fact (pred c_1)))) =

= if iszero c_3 then c_1 else times c_3 (if iszero c_2 then c_1 else times c_2 ->

-> (if iszero c_1 then c_1 else times c_1 (fact c_0))) =

= if iszero c_3 then c_1 else times c_3 (if iszero c_2 then c_1 else times c_2 ->

-> (if iszero c_1 then c_1 else times c_1 ->

-> (if $\underbrace{\text{iszero } c_0}_{\text{True}}$ then c_1 else times c_0 (fact (pred c_0)))) =

= if iszero c_3 then c_1 else times c_3 (if iszero c_2 then c_1 else times c_2 ->

-> (if iszero c_1 then c_1 else times c_1 c_1) =

= if iszero c_3 then c_1 else times c_3 (if iszero c_2 then c_1 else times c_2 ->

-> (if $\underbrace{\text{iszero } c_1}_{\text{False}}$ then c_1 else c_1) =

= if iszero c_3 then c_1 else times c_3 (if iszero c_2 then c_1 else times c_2 c_1) =

= if iszero c_3 then c_1 else times c_3 (if $\underbrace{\text{iszero } c_2}_{\text{False}}$ then c_1 else c_2) =

= if iszero c_3 then c_1 else times c_3 c_2 =

= if $\underbrace{\text{iszero } c_3}_{\text{False}}$ then c_1 else c_6 =

= c_6