Week 7. Polymorphic type theory

In-class problems

1. Give a proof in minimal logic of the formula

$$\forall c.\, (\forall c.\, c) \to c$$

and give the corresponding proof term and its type in $\lambda 2$ notation.

2. Define the notation

$$A \vee B \quad := \quad \forall c.\, (A \to c) \to (B \to c) \to c$$

Give terms that correspond to the deduction rules for disjunction, i.e., fill the '?' in:

$$\begin{aligned} a : *,\ b : * \ &\vdash\ ? : a \to a \vee b \\ a : *,\ b : * \ &\vdash\ ? : b \to a \vee b \\ a : *,\ b : *,\ c : * \ &\vdash\ ? : a \vee b \to (a \to c) \to (b \to c) \to c \end{aligned}$$

3. Define (in $\lambda 2P$):

$$\mathsf{eq} \quad := \quad \lambda a : *.\, \lambda x : a.\, \lambda y : a.\, \Pi p : a{\to}*.\, p\, x \to p\, y$$

This function is called Leibniz equality. What is the type of eq, i.e., what is '?' in

$$\vdash \quad \mathsf{eq} : ?$$

Show that eq is an equivalence relation, i.e., find inhabitants

$$\begin{aligned} a : *,\ x : a \ &\vdash\ ? : \mathsf{eq}\, a\, x\, x \\ a : *,\ x : a,\ y : a \ &\vdash\ ? : \mathsf{eq}\, a\, x\, y \to \mathsf{eq}\, a\, y\, x \\ a : *,\ x : a,\ y : a,\ z : a \ &\vdash\ ? : \mathsf{eq}\, a\, x\, y \to \mathsf{eq}\, a\, y\, z \to \mathsf{eq}\, a\, x\, z \end{aligned}$$

Also show that eq has the substitution property, i.e., inhabit:

$$a : *,\ x : a,\ y : a,\ p : a{\to}* \ \vdash\ ? : \mathsf{eq}\, a\, x\, y \to p\, x \to p\, y$$

4. Are there long normal forms that solve the following inhabitation problems? Explain your answers.

$$\begin{aligned} a : *,\ f : * \to * \ &\vdash\ ? : (fa \to a) \to (a \to fa) \to f(fa) \to a \\ a : *,\ f : * \to * \ &\vdash\ ? : (\forall a.\, fa \to a) \to f(fa) \to a \end{aligned}$$

Take-home problems

1. Give a full derivation of the type judgment

$$b : * \;\vdash\; \Pi a.\, a \to b : *$$

2. Given the type of polymorphic Church numerals

$$\mathsf{nat} \;:\; \forall a.\, (a \to a) \to a \to a$$

define truncated subtraction on this type, i.e., define

$$\mathsf{sub} \;:\; \mathsf{nat} \to \mathsf{nat} \to \mathsf{nat}$$

such that for polymorphic Church numerals c_n and c_m holds that

$$\mathsf{sub}\ c_n\ c_m \twoheadrightarrow_\beta \begin{cases} c_{m-n} & \text{if } m \geq n \\ c_0 & \text{otherwise} \end{cases}$$

3. Give an impredicative definition for the data-type $\mathsf{tree}\ A$ of binary trees with elements of $A : *$ at the leaves. Define functions

$$\begin{aligned} \mathsf{leaf} &\;:\; \Pi a : *.\, a \to \mathsf{tree}\ a \\ \mathsf{node} &\;:\; \Pi a : *.\, \mathsf{tree}\ a \to \mathsf{tree}\ a \to \mathsf{tree}\ a \end{aligned}$$

as constructors and

$$\mathsf{tree\_rec} \;:\; \Pi b : *.\, (a \to b) \to (\mathsf{tree}\ a \to b \to \mathsf{tree}\ a \to b \to b) \to \mathsf{tree}\ a \to b$$

as recursor. Given $\mathsf{nat} : *$ and $\mathsf{plus} : \mathsf{nat} \to \mathsf{nat} \to \mathsf{nat}$, use the recursor to define a function that sums the elements in the tree:

$$\mathsf{sum\_tree} \;:\; \mathsf{tree\ nat} \to \mathsf{nat}$$

4. Compare the impredicative definitions of the various logical operations with the customary elimination rules of these operations. For the operations for which there is no direct correspondence, give an alternative elimination rule that *does* correspond to the impredicative definition. Show that these alternative elimination rules are equivalent to the normal ones, in the sense that if you have one, you can simulate the other.