In-class problems

The following exercises are about the system λP. In the first three exercises, let $\Gamma = \{A : *, P : A \to *\}$.

1. Find a term ? such that

$$\Gamma \vdash ? : \Pi a{:}A.Pa \to Pa$$

2. Let $\Phi = \lambda f : A{\to}A.\ \Pi a{:}A.Pa{\to}P(fa)$. Find a type ? such that

$$\Gamma \vdash \Phi : ?$$

3. Give a term M and a complete derivation tree of

$$\Gamma \vdash M : \Phi(\lambda a{:}A.a)$$

4. Let $\neg A = A{\to}o$. Prove that any antisymmetric relation is irreflexive:

   $A{:}*, R{:}A{\to}A{\to}*, \mathtt{asym}{:}\,\Pi x{:}A.\Pi y{:}A.Rxy \to \neg Ryx \ \vdash\ ?\ : \Pi x{:}A.\neg Pxx$

5. Given the following λP context $\Gamma :=$

$$\begin{array}{l} \mathsf{nat} : *,\ 0 : \mathsf{nat},\ 1 : \mathsf{nat},\ \mathsf{plus} : \mathsf{nat} \to \mathsf{nat} \to \mathsf{nat}, \\ \mathsf{vec} : \mathsf{nat} \to *, \\ \mathsf{nil} : \mathsf{vec}\ 0, \\ \mathsf{cons} : \Pi n : \mathsf{nat}.\, \mathsf{nat} \to \mathsf{vec}\ n \to \mathsf{vec}\ (\mathsf{plus}\ n\ 1) \end{array}$$

   In this vec represents vectors of natural numbers of a given length, nil is the empty vector, and cons adds one element to the front of a vector.

   (a) Give the term in this context that represents the vector $\langle 1, 2, 3\rangle$.

   (b) Give types for the function append that concatenates two vectors, and for the function $\mathsf{reverse}$ that reverses a vector.

   (c) In the context

$$\Gamma,\ n : \mathsf{nat},\ m : \mathsf{nat},\ l : \mathsf{vec}\ n, k : \mathsf{vec}\ m$$

   give two terms that represent the same vector, the reverse of the concatenation of l and k, one by first appending and then reversing, and one by first reversing and then concatenating. What are the types of those two terms? Are they the same?

Take-home problems

1. Give a proof in minimal predicate logic of

$$(\forall x\, y\, z.\, r(x,y) \to r(x,z) \to r(y,z)) \to (\forall x.\, r(x,x)) \to \\ (\forall x\, y.\, r(x,y) \to r(y,x))$$

   give its proof term, and give the type judgment of the proof term.

2. The context

   $\mathsf{prop} : *,$
   $\mathsf{imp} : \mathsf{prop} \to \mathsf{prop},$
   $\mathsf{proof} : \mathsf{prop} \to \mathsf{proof},$
   $\mathsf{imp\_i} : \forall A : \mathsf{prop}.\, \forall B : \mathsf{prop}.\, (\mathsf{proof}\ A \to \mathsf{proof}\ B) \to \mathsf{proof}\ (\mathsf{imp}\ A\ B),$
   $\mathsf{imp\_e} : \forall A : \mathsf{prop}.\, \forall B : \mathsf{prop}.\, \mathsf{proof}\ (\mathsf{imp}\ A\ B) \to \mathsf{proof}\ A \to \mathsf{proof}\ B$

   gives another way to encode logic following the Curry-Howard isomorphism. Here the logic does not need to fit the 'intrinsic' logic of λP but can be many different logics. This is called using λP as a *logical framework*.

   (a) In this context, give the term that corresponds to a proof of $a \to b \to a$.
   (b) Extend this context with the introduction and elimination rules of conjunction.
   (c) In this extended context, give the term that corresponds to a proof of $a \land b \to b \land a$.

3. Give a context that encodes untyped combinatory logic, including its theory of equality, as a logical framework.

4. Consider the following six types

$$\Pi x : a.\, b \quad \Pi x : a.\, p\, x \quad \Pi a : *.\, b \quad \Pi a : *.\, a \quad \Pi x : a.\, * \quad \Pi a : *.\, *$$

   where $a : *$, $b : *$ and $p : a \to *$.

   (a) Which of these types can also be written with $\to$ notation, instead of using Π? For the types that can be written that way, write them using $\to$ notation.
   (b) In which systems of the λ-cube are each of these six types allowed?
   (c) What are the types of these six types in the systems of the λ-cube?