EB 3 - Let-Polymorphism

Exercise 1

Suppose the condition $s \notin \Gamma$ below is dropped in the ML1 rule T-TAbs:

$$\frac{\Gamma \rhd M : \pi \quad s \notin \Gamma}{\Gamma \rhd \Lambda s.M : \forall s.\pi}\ \text{T-TAbs}$$

Show that one can then type a function that casts any type to any other.

Exercise 2

Exhibit typing derivations in ML1 for:

I. $\mathsf{let}\ g : \forall s.s \to \mathbb{N} = \lambda x : s{,}5\ \mathsf{in}\ \langle g\ \mathbb{B}\ True, g\ \mathbb{N}\ 3\rangle$

II. $\mathsf{let}\ g : \forall s.s \to s = \lambda x : s.x\ \mathsf{in}\ \langle g\ \mathbb{B}\ True, g\ \mathbb{N}\ 3\rangle$

Exercise 3

Show that the judgement $x : t \rhd (\Lambda t.x)\,\mathbb{N} : \mathbb{N}$ is not derivable in ML1. Show that, if it were the case, then type preservation would fail.

Exercise 4

Exhibit typing derivations in ML2 for the following judgements:

I. $\mathsf{let}\ g : \forall s.s \to \mathbb{N} = \lambda x : s{,}5\ \mathsf{in}\ \langle g\ True, g\ 3\rangle$

II. $\mathsf{let}\ d : \forall s.(s \to s) \to s = \lambda f : s \to s.\lambda x : s.f\ (f\ x)\ \mathsf{in}\ \langle d\ (+1)\ 2, d\ not\ True\rangle$

Exercise 5

Exhibit two different types σ_1 and σ_2 s.t. $\mathsf{let}\ x = \lambda z : s.z\ \mathsf{in}\ x$ is typable with both types in ML2.

Exercise 6

Prove that $\Gamma \rhd_2 M : \sigma \Rightarrow \exists M'.\text{EraseT}(M') = M \wedge \Gamma \rhd M' : \sigma$ where

$$\begin{aligned}
\text{EraseT}(x) &\stackrel{\text{def}}{=} x \\
\text{EraseT}(true) &\stackrel{\text{def}}{=} true \text{ same with other constants} \\
\text{EraseT}(if\ M\ then\ P\ else\ Q) &\stackrel{\text{def}}{=} if\ \text{EraseT}(M)\ then\ \text{EraseT}(P)\ else\ \text{EraseT}(Q) \\
\text{EraseT}(\lambda x : \sigma.M) &\stackrel{\text{def}}{=} \lambda x : \sigma.\text{EraseT}(M) \\
\text{EraseT}(M\,N) &\stackrel{\text{def}}{=} \text{EraseT}(M)\,\text{EraseT}(N) \\
\text{EraseT}(\Lambda s.M) &\stackrel{\text{def}}{=} \text{EraseT}(M) \\
\text{EraseT}(M\,\sigma) &\stackrel{\text{def}}{=} \text{EraseT}(M) \\
\text{EraseT}(\mathsf{let}\ x : \pi = M\ \mathsf{in}\ N) &\stackrel{\text{def}}{=} \mathsf{let}\ x : \pi = \text{EraseT}(M)\ \mathsf{in}\ \text{EraseT}(N)
\end{aligned}$$

Exercise 7

Infer the type of the following expressions:

I. $\mathsf{let}\ i = \lambda x.x\ \mathsf{in}\ i\,i$

II. $\mathsf{let}\ g = \lambda x{,}5\ \mathsf{in}\ \langle g\ true, g\ 3\rangle$

III. $\lambda y.\mathsf{let}\ x = \lambda z.z\,y\ \mathsf{in}\ \langle x\,(\lambda z.succ(z)), x\,(\lambda z.isZero(z))\rangle$

Exercise 8

Show that the term:

$$\begin{aligned}
\mathsf{letrec}\ f \quad = \quad & \lambda i : \mathbb{N}.\lambda x : s. \\
& \quad if\ isZero(i)\ then\ x \\
& \quad else\ f\ pred(i)\ \langle x, x\rangle \\
\mathsf{in}\ (f\ 87)\ true &
\end{aligned}$$

is not typable with the typing rule:

$$\frac{\Gamma, x : \tau \rhd_2 M : \tau \qquad \Gamma, x : \mathsf{gen}(\tau, \Gamma) \rhd_2 N : \sigma}{\Gamma \rhd_2 \mathsf{letrec}\ x : \mathsf{gen}(\tau, \Gamma) = M\ \mathsf{in}\ N : \sigma}\ \textsc{T-Letrec}$$

Exercise 9

Consider the following *typing declarations* of polymorphic constants:

$$\begin{array}{rcl} Nil & :: & \forall s. List\ s \\ Cons & :: & \forall s. s \to List\ s \to List\ s \\ caseL & :: & \forall s. \forall t. List\ s \to t \to (s \to List\ s \to t) \to t \end{array}$$

Assume you have the typing rule for polymorphic constants:

$$\frac{c : \forall \vec{s}. \tau}{\Gamma \rhd_2 c : \tau\{\vec{s} \leftarrow \vec{\sigma}\}}\ \textsc{Cst}$$

I. Type $Cons\ 1\ (Cons\ 2\ (Cons\ 3\ Nil)) :: List\ \mathbb{N}$.

II. Provide the type of a polymorphic constant for *map* and type

$$\lambda z : List\,\mathbb{N}. map\ (\lambda x : \mathbb{N}. isZero(x))\ (map\ (\lambda x : \mathbb{N}. succ(x))\ z).$$