Types 2005 – Paper 7 Question 9 (AMP)

Value restricted typing rule:

2 $$(\text{letv})\ \frac{\Gamma \vdash M_1 : \tau_1 \qquad \Gamma, x : \forall A(\tau_1) \vdash M_2 : \tau_2}{\Gamma \vdash \text{let } x = M_1 \text{ in } M_2 : \tau_2}$$

provided $x \notin dom(\Gamma)$

2 and $A = \begin{cases} \{\} & \text{if } M_1 \text{ is not a value} \\ ftv(\tau_1) - ftv(\Gamma) & \text{if } M_1 \text{ is a value} \end{cases}$

where $ftv(\ldots)$ = free type variables of...

1 and values are $V ::= x \mid \lambda x(M) \mid () \mid true \mid false \mid \cdots$

5

Apart from (letv), we use the following typing rules

(var≻) $\Gamma \vdash x : \tau$ if $x \in dom(\Gamma)$ & $\Gamma(x) \succ \tau$

(bool) $\Gamma \vdash true : bool$

(fn) $\dfrac{\Gamma, x : \tau_1 \vdash M : \tau_2}{\Gamma \vdash \lambda x(M) : \tau_1 \to \tau_2}$ if $x \notin dom(\Gamma)$

(app) $\dfrac{\Gamma \vdash M : \tau_1 \to \tau_2 \qquad \Gamma \vdash M' : \tau_1}{\Gamma \vdash M M' : \tau_2}$

(unit) $\Gamma \vdash () : unit$

(ref) $\dfrac{\Gamma \vdash M : \tau}{\Gamma \vdash ref\, M : \tau\, ref}$

(get) $\dfrac{\Gamma \vdash M : \tau\, ref}{\Gamma \vdash\ !M : \tau}$ and (set) $\dfrac{\Gamma \vdash M_1 : \tau\, ref \qquad \Gamma \vdash M_2 : \tau}{\Gamma \vdash M_1 := M_2 : unit}$

and write $\Gamma \vdash M : \forall A(\tau)$ if $\Gamma \vdash M : \tau$ with $A = ftv(\tau) - ftv(\Gamma)$.

1 (i) <u>Not provable.</u>

1 For if it were, since ref $\lambda x(x)$ is <u>not</u> a value, it would have been derived using (letv) from

1 (1) $\{\} \vdash \text{ref } \lambda x(x) : \tau_1$

(2) $\{r : \tau_1\} \vdash (!r)(r := \lambda y(\text{true})) : \text{unit}$

for some τ_1.

This instance of r has type $(\text{unit} \to \tau_2)\text{ ref}$, some τ_2

This instance of r has type $(\tau_3 \to \text{bool})\text{ ref}$, some τ_3

So we would have to have

$$\tau_1 > (\text{unit} \to \tau_2)\text{ ref} \ \& \ \tau_1 > (\tau_3 \to \text{bool})\text{ ref}$$

i.e. $(\text{unit} \to \tau_2)\text{ ref} = \tau_1 = (\tau_3 \to \text{bool})\text{ ref}$

2 so that $\tau_2 = \text{bool}$ & $\tau_3 = \text{unit}$ & $\tau_1 = (\text{unit} \to \text{bool})\text{ ref}$.

But then (1) cannot hold, since if it did, we have to have proved $\{\} \vdash \lambda x(x) : \text{unit} \to \text{bool}$,

5 which does not follow from the typing rules.

1 (ii) <u>Is provable.</u> Putting $\Gamma = \{r : (\text{unit} \to \text{unit})\text{ref}\}$, we have

$$\dfrac{\dfrac{\dfrac{}{\Gamma \vdash r : (\text{unit} \to \text{unit})\text{ref}}\,(\text{var})}{\Gamma \vdash\, !r : \text{unit} \to \text{unit}}\,(\text{get}) \qquad \dfrac{\dfrac{}{\Gamma \vdash r : (\text{unit} \to \text{unit})\text{ref}}\,(\text{var}) \quad \dfrac{\dfrac{}{\cdots}\,(\text{var})}{\Gamma \vdash \lambda y() : \text{unit} \to \text{unit}}\,(\text{fn})}{\Gamma \vdash r := \lambda y(()) : \text{unit}}\,(\text{set})}{\Gamma \vdash (!r)(r := \lambda y()) : \text{unit}}\,(\text{app})$$

1

and

$$\dfrac{\dfrac{\dfrac{}{\{x : \text{unit}\} \vdash x : \text{unit}}\,(\text{var})}{\{\} \vdash \lambda x(x) : \text{unit} \to \text{unit}}\,(\text{fn})}{\{\} \vdash \text{ref } \lambda x(x) : (\text{unit} \to \text{unit})\text{ref}}\,(\text{ref})$$

1

3 So we can apply (letv) to deduce (ii).

2 (iii) Is provable, with $\sigma = \forall\alpha((\alpha \to \alpha\,\text{ref})\,\text{ref})$

For we have

$$\dfrac{\dfrac{\dfrac{}{\{x:\alpha\} \vdash x:\alpha}(\text{var}\succ)}{\{x:\alpha\} \vdash \text{ref}\,x : \alpha\,\text{ref}}(\text{ref})}{\{\} \vdash \lambda x(\text{ref}\,x) : \alpha \to \alpha\,\text{ref}}(\text{fn})$$

and putting $\Gamma = \{f : \forall\alpha(\alpha \to \alpha\,\text{ref})\}$, we have

1 $$\dfrac{\dfrac{}{\Gamma \vdash f : (\alpha \to \alpha\,\text{ref}) \to (\alpha \to \alpha\,\text{ref})\,\text{ref}}(\text{var}\succ) \qquad \dfrac{}{\Gamma \vdash f : \alpha \to \alpha\,\text{ref}}(\text{var}\succ)}{\Gamma \vdash f\,f : (\alpha \to \alpha\,\text{ref})\,\text{ref}}$$

so we can apply (letv) to deduce

$$\{\} \vdash \text{let}\ f = \lambda x(\text{ref}\,x)\ \text{in}\ f\,f \ :\ (\alpha \to \alpha\,\text{ref})\,\text{ref}$$

③ & hence (iii) with σ as above.

---

1 (iv) Is provable :

$$\dfrac{\dfrac{\dfrac{}{\{x:\alpha, f:\alpha \to \beta\} \vdash f : \alpha \to \beta}(\text{var}\succ) \qquad \dfrac{}{\{x:\alpha, f:\alpha \to \beta\} \vdash x : \alpha}(\text{var}\succ)}{\{x:\alpha, f:\alpha \to \beta\} \vdash f\,x : \beta}(\text{app})}{\{x:\alpha\} \vdash \lambda f(f\,x) : (\alpha \to \beta) \to \beta}(\text{fn})$$

② 1 and since $ftv((\alpha \to \beta) \to \beta) - ftv(\{x:\alpha\}) = \{\beta\}$, we get (iv).

---

1 (v) Is not provable.

If it were, we'd have to have proved

$$\{x:\beta\} \vdash \lambda f(f\,x) : (\beta' \to \beta') \to \beta'$$

with $\beta' \neq \beta$, from (fn) and

$$\{x:\beta, f:\beta' \to \beta'\} \vdash f\,x : \beta'$$

1 and the latter from $\begin{cases} \{x:\beta, f:\beta' \to \beta'\} \vdash f : \tau \to \beta', \text{ some } \tau \\ \{x:\beta, f:\beta' \to \beta'\} \vdash x : \tau \end{cases}$

② but there is no τ that makes these provable from (var$\succ$).

Commentary

The ML type system with value-restricted let-rule is covered in Lecture 4. Examples like (but not the same as) (i)–(iii) were given in the Lectures. Examples (iv) & (v) test the students' understanding of when & how type variables can be generalised to assign a type scheme (rather than just a type) to an ML expression — covered in Lectures 2 & 3.