Programmation 1

TD n°12

8 décembre 2020

1 Real PCF$^-$

We give below the denotational and operational semantics for Real PCF$^-$.
The types are as follows :

$$\begin{aligned}\sigma, \tau, \ldots ::= &\ \texttt{unit} \\ |&\ \texttt{I} \\ |&\ \sigma \to \tau\end{aligned}$$

$\mathbb{S} = \{\bot, \top\}$ with $\bot < \top$. $\mathcal{I} = [0,1]$ with the usual order.

$$[\![\texttt{unit}]\!] = \mathbb{S} \qquad [\![\texttt{I}]\!] = \mathcal{I} \qquad [\![\sigma \to \tau]\!] = [[\![\sigma]\!] \to [\![\tau]\!]].$$

$$\begin{aligned}
[\![x_\tau]\!]\rho &= \rho(x_\tau) \\
[\![uv]\!]\rho &= [\![u]\!]\rho([\![v]\!]\rho) \\
[\![\texttt{fn } x_\sigma.u]\!]\rho &= (V \in [\![\sigma]\!] \mapsto [\![u]\!](\rho[x_\sigma \mapsto V])) \\
[\![\texttt{letrec } x_\sigma = u \texttt{ in } v]\!]\rho &= [\![v]\!](\rho[x_\sigma \mapsto [\![\texttt{rec } x_\sigma = u]\!]\rho]) \\
[\![\texttt{rec } x_\sigma = u]\!]\rho &= \mathrm{lfp}(V \in [\![\sigma]\!] \mapsto [\![u]\!](\rho[x_\sigma \mapsto V])) \\
[\![0.u]\!]\rho &= add_0([\![u]\!]\rho) \qquad\qquad [\![1.u]\!]\rho = add_1([\![u]\!]\rho) \\
[\![\texttt{tl}_0 u]\!]\rho &= rem_0([\![u]\!]\rho) \qquad\qquad [\![\texttt{tl}_1 u]\!]\rho = rem_1([\![u]\!]\rho) \\
[\![\texttt{pif } u \texttt{ then } v \texttt{ else } w : \tau]\!]\rho &= \begin{cases} [\![v]\!]\rho, & \text{if } [\![u]\!]\rho = \top \\ [\![v]\!]\rho \wedge [\![w]\!]\rho, & \text{if } [\![u]\!]\rho = \bot \end{cases} \\
[\![u > \texttt{1/2}]\!]\rho &= \begin{cases} \top, & \text{if } [\![u]\!]\rho > 1/2 \\ \bot, & \text{otherwise} \end{cases} \qquad [\![u > \texttt{0}]\!]\rho = \begin{cases} \top, & \text{if } [\![u]\!]\rho > 0 \\ \bot, & \text{otherwise} \end{cases} \\
[\![*]\!]\rho &= \top,
\end{aligned}$$

where $V \in X \mapsto f(V)$ denotes the function which to all V in X associates $f(V)$, and where :

$$\begin{aligned}
add_0(a) &= a/2 \qquad\qquad add_1(a) = (a+1)/2 \\
rem_0(a) &= \min(2a, 1) \qquad\qquad rem_1(a) = \max(2a-1, 0)
\end{aligned}$$

Contexts (type constraints omitted) :

$$\begin{aligned}\mathcal{C} ::= {} & \_ \\ & \mid \mathcal{C}v \\ & \mid \texttt{tl}_0\mathcal{C} \\ & \mid \texttt{tl}_0\mathcal{C} \\ & \mid \texttt{tl}_1\mathcal{C} \\ & \mid \mathcal{C} > \texttt{1/2} \\ & \mid \mathcal{C} > \texttt{0} \\ & \mid \texttt{pif } \mathcal{C} \texttt{ then } v \texttt{ else } w \\ & \mid \texttt{pif } u \texttt{ then } \mathcal{C} \texttt{ else } w \\ & \mid \texttt{pif } u \texttt{ then } v \texttt{ else } \mathcal{C}\end{aligned}$$

Operational semantics. We only apply a rule under a context $\mathcal{C}$ of the above form, i.e., $u \to v$ if and only if $u = \mathcal{C}[\ell]$ and $v = \mathcal{C}[r]$, where $\mathcal{C}$ is a context (the types being respected), and $\ell \to r$ is one of the rules below.

$$\begin{aligned}
(\texttt{fn } x_\sigma.u)v &\to u[x_\sigma := v] \\
\texttt{letrec } x_\sigma = u \texttt{ in } v &\to v[x_\sigma := \texttt{letrec } x_\sigma = u \texttt{ in } v] \\
\texttt{tl}_a(a.u) &\to u && (a \in \{0,1\}) \\
\texttt{tl}_0(1.u) &\to \dot{1} \\
\texttt{tl}_1(0.u) &\to \dot{0} \\
(1.u) > \texttt{1/2} &\to u > 0 \\
(1.u) > 0 &\to * \\
(0.u) > 0 &\to u > 0 \\
\texttt{pif } * \texttt{ then } v \texttt{ else } w &\to v \\
\texttt{pif } u \texttt{ then } v \texttt{ else } * &\to v && (\alpha) \\
\texttt{pif } u \texttt{ then } 0.v \texttt{ else } 1.w &\to 0.v \\
\texttt{pif } u \texttt{ then } a.v \texttt{ else } a.w &\to a.(\texttt{pif } u \texttt{ then } v \texttt{ else } w) && (a \in \{0,1\})
\end{aligned}$$

Exercise 1 :

Recall that for all $u : \tau, [\![u]\!]$ is a well-defined function, Scott-continuous from $Env \stackrel{\text{def}}{=} \prod_{x_\sigma \text{variable}} [\![\sigma]\!]$ to $[\![\tau]\!]$.

1. Show that the construction $u > 0$ of Real PCF$^-$ is redundant. Explicitly propose a definition of an expression Real PCF$^-$ **nonzero**, of type $\texttt{I} \to \texttt{unit}$, which does not use the expression of the form $u > 0$, and whose semantics $[\![\textbf{nonzero}]\!]\rho$ is the function to which 0 associates $\perp$ and to all $a \in \mathcal{I}$ non-zero associates $\top$. Prove this assertion.
2. Show that the rule tagged with (α) of the operational semantics is correct, in the sense that $[\![\texttt{pif } u \texttt{ then } v \texttt{ else } *]\!]\rho = [\![v]\!]\rho$ for all $\rho \in Env$.
3. We consider a Real PCF$^-$ program of the form $\texttt{letrec } x_\sigma = u \texttt{ in } v$, of type `unit` . Show that if $[\![\texttt{letrec } x_\sigma = u \texttt{ in }]\!]\rho \neq \perp$,then there is an integer $n \in \mathbb{N}$ such that

$$[\![\texttt{letrec } x_\sigma = u \texttt{ in }]\!]\rho = g(f^n(\perp)),$$

where we use the abbreviations $g(V) = [\![v]\!](\rho[x_\sigma \mapsto V])$ and $f(V) = [\![u]\!](\rho[x_\sigma \mapsto V])$. (The $\perp$ in argument of f^n is that of $[\![\sigma]\!]$.) This expresses that a recursive definition (of x_σ) used in a terminating computation (v) of type `unit` will be "expanded" only n times .

4. Why does the argument from the previous question not work if $\texttt{letrec}\ x_\sigma = u\ \texttt{in}\ v$ is of type $\texttt{I}$?
5. Recall that $\dot{0} \stackrel{\text{def}}{=} \texttt{letrec}\ x_{\texttt{I}} = 0.x_{\texttt{I}}\ \texttt{in}\ x_{\texttt{I}}$. Show that there does not exist a derivation in the operational semantics for

$$\texttt{t1}_0(\texttt{pif}\ \dot{0} > \texttt{1/2 then}\ 1.\dot{0}\ \texttt{else}\ 0.1.1.\dot{0}) > \texttt{1/2} \to^* *.$$

   We can set $Z \stackrel{\text{def}}{=} \texttt{letrec}\ x_{\texttt{I}} = 0.x_{\texttt{I}}\ \texttt{in}\ 0.x_{\texttt{I}}$.
6. What can we conclude for the adequacy of the type $\texttt{unit}$? Justify.
7. Any suggestions to complete the operational semantics ?

Exercise 2 :

We now assume that a same Real PCF$^-$ variable is always labeled with the same type : if we see x_σ and x_τ, then $\sigma = \tau$. This amounts to saying that the name x of the variables is sufficient to distinguish them.

We consider the Real PCF^{--} language, which is just Real PCF$^-$ but without any type index. For example, $\texttt{fn}\ x.u$ and $\texttt{letrec}\ x = u\ \texttt{in}\ v$ are the expressions Real PCF^{--} corresponding to $\texttt{fn}\ x_\sigma.u$ and $\texttt{letrec}\ x_\sigma = u\ \texttt{in}\ v$, respectively.

Formally, let E denote the type erasure function, defined by $E(\texttt{letrec}\ x_\sigma = u\ \texttt{in}\ v) \stackrel{\text{def}}{=} \texttt{letrec}\ x = E(u)\ \texttt{in}\ E(v)$, $E(\texttt{fn}\ x_\sigma.u) \stackrel{\text{def}}{=} \texttt{fn}\ x.E(u)$, etc.

We will say that a Real PCF^{--} expression u is typable, of type τ, if and only if there exists a Real PCF$^-$ expression u', of type τ, such that $E(u) = u$.

1. Are all Real PCF^{--} expressions typable ? Justify.
2. Is the type of a Real PCF^{--} typable expression unique ? Justify.