λ_{eff}

nymphium

January 5, 2019

Syntax

$$
\begin{aligned}
x &\in& Variables \\
eff &\in& Effects
\end{aligned}
$$

$$
\begin{aligned}
v &::=& x \mid h \mid \lambda x.e \mid eff \\
e &::=& v \mid e\,e \mid \texttt{let } x = e \texttt{ in } e \\
&\mid& \texttt{inst } () \mid \texttt{with } v \texttt{ handle } e \\
&\mid& \texttt{perform } e\,e \\
h &::=& \texttt{handler } v\ (\texttt{val } x \to e)\,((x,k) \to e) \\[4pt]
F &::=& e\,\square \mid \square\,v \mid \texttt{let } x = \square \texttt{ in } e \\
&\mid& \texttt{with } v \texttt{ handle } \square \mid \texttt{perform } \square\,e \mid \texttt{perform } v\,\square \\
s &::=& [] \mid F :: s
\end{aligned}
$$

Figure 1: the syntax of λ_{eff}

Semantics

$$
\begin{aligned}
\textit{flatfn } [] &= \lambda x.x \\
\textit{flatfn } (F :: s) &= \lambda x.\textit{flatfn } s\ (F\,[x])
\end{aligned}
$$

Figure 2: utils for the semanticcs

$$
\langle e;\, s;\, es \rangle \mapsto \langle e';\, s';\, es' \rangle
$$

$$
\langle F\,[e];\, s;\, es \rangle \mapsto \langle e;\, F :: s;\, es \rangle \tag{Push}
$$

$$
\langle v;\, F :: s;\, es \rangle \mapsto \langle F\,[v];\, s;\, es \rangle \tag{Pop}
$$

$$
\langle v;\, [];\, es \rangle \mapsto \langle v;\, [];\, es \rangle \tag{Result}
$$

$$
\langle \lambda x.e;\, (\square\,v) :: s;\, es \rangle \mapsto \langle e\,[x = v];\, s;\, es \rangle \tag{Apply}
$$

$$
\langle \texttt{inst } ();\, s;\, es \rangle \mapsto \langle eff;\, s;\, es \rangle \tag{Instanciate}
$$

$$
\langle \texttt{perform } eff\,v;\, F :: s;\, es \rangle \mapsto \langle \texttt{perform } eff\,v;\, s;\, F :: es \rangle \tag{Rethrow}
$$

$$
\begin{aligned}
\langle \texttt{perform } eff\,v;\, F :: s;\, es \rangle &\mapsto \langle e_{eff}\,[x = v, k = \textit{flatfn } es];\, F :: s;\, [] \rangle \\
\text{where} \quad F &= \texttt{with } h \texttt{ handle } \square \\
h &= \texttt{handler } eff\ (\texttt{val } x \to e_v)\,((x,k) \to e_{eff})
\end{aligned} \tag{Handle$_{\text{EFF}}$}
$$

$$
\langle v;\, (\texttt{handler } eff\ (\texttt{val } x \to e_v)\,((x,k) \to e_{eff})) :: s;\, es \rangle \mapsto \langle e_v[x = v];\, s;\, es \rangle \tag{Handle$_{\text{V}}$}
$$

$$
\langle \texttt{perform } eff\,v;\, [];\, es \rangle \mapsto abort \tag{Leak}
$$

Figure 3: the semantics of λ_{eff}