Selam specification

nymphium

August 15, 2019

1 syntax

$$
\begin{array}{rcl}
x & \in & \mathit{Variables} \\
a & \in & \mathit{Atomics} \\
v & ::= & a \mid \text{¡}\lambda x.e \mid \text{¡}x \mid \lambda x.e \mid x \\
e & ::= & v \mid e@e \mid \texttt{¡let}\ x = e\ \texttt{in}\ e \\
 & & \mid e\ e \mid \texttt{let}\ x = e\ \texttt{in}\ e \\
 & & \\
w & ::= & a \mid \texttt{¡clos}\,(x, e, E, \text{¡}E) \mid \texttt{clos}\,(\lambda x.e, E) \\
F & ::= & w\ \square \mid w@\square \mid \texttt{¡app}\,(\square, e, E, \text{¡}E) \mid \texttt{app}\,(\square, e, E) \\
 & & \mid \texttt{let}_x\,(e, E) \mid \texttt{¡let}_x\,(e, E, \text{¡}E) \\
C & ::= & e \mid w \\
E & ::= & [] \mid (x = w)\,, E \\
\text{¡}E & ::= & [] \mid (x = w)\,, \text{¡}E \\
K & ::= & [] \mid F, K \\
 & & \\
\tau & ::= & A \mid \tau \to \tau \mid \tau \multimap \tau \\
\Gamma & ::= & [] \mid (x = \tau)\,, \Gamma \\
\Delta & ::= & [] \mid (x = \tau)\,, \Delta
\end{array}
$$

2 semtntics

$$\boxed{\langle C;\ E;\ \text{¡}E;\ K\rangle \mapsto \langle C';\ E';\ \text{¡}E';\ K'\rangle}$$

$$\langle x;\ E;\ \text{¡}E;\ K\rangle \mapsto \langle \mathit{lookup}\,(x, E)\,;\ E;\ \text{¡}E;\ K\rangle \qquad (\text{Lookup})$$

$$\langle \text{¡}x;\ E;\ \text{¡}E;\ K\rangle \mapsto \langle \mathit{lookup}\,(\text{¡}x, \text{¡}E)\,;\ E;\ \mathit{drop}\,(\text{¡}x, \text{¡}E)\,;\ K\rangle \qquad (\text{¡Lookup})$$

$$\langle \texttt{let}\ x = e\ \texttt{in}\ e';\ E;\ \text{¡}E;\ K\rangle \mapsto \langle e;\ E;\ \text{¡}E;\ (\texttt{let}_x\,(e', E)\,, K)\rangle \qquad (\text{PushLet})$$

$$
\begin{aligned}
lookup\,(x, []) &= \text{undefined}\\
lookup\,(x, ((x = v)\,, \_)) &= v\\
lookup\,(x, ((y = \_)\,, ls)) &= lookup\,(x, ls)\\
\\
drop\,(x, []) &= []\\
drop\,(x, ((x = \_)\,, ls)) &= ls\\
drop\,(x, ((y = v)\,, ls)) &= (y = v)\,, drop\,(x, ls)
\end{aligned}
$$

Figure 1: utility functions

$$\langle \mathtt{ilet}\ x = e\ \mathtt{in}\ e';\ E;\ \mathsf{i}E;\ K\rangle \mapsto \langle e;\ E;\ \mathsf{i}E;\ (\mathtt{ilet}_x\,(e', E, \mathsf{i}E)\,, K)\rangle \quad (\textsc{PushiLet})$$

$$\langle w;\ E;\ \mathsf{i}E;\ \mathtt{let}_x\,(e', E')\,, K\rangle \mapsto \langle e';\ ((x = w)\,, E')\,;\ \mathsf{i}E;\ K\rangle \quad (\textsc{PopLet})$$

$$\langle w;\ E;\ \mathsf{i}E;\ (\mathtt{ilet}_x\,(e, E', \mathsf{i}E')\,, K)\rangle \mapsto \langle e;\ E';\ ((x = w)\,, \mathsf{i}E')\,;\ K\rangle \quad (\textsc{PopiLet})$$

$$\langle \lambda x.e;\ E;\ \mathsf{i}E;\ K\rangle \mapsto \langle \mathtt{clos}\,(\lambda x.e, E)\,;\ E;\ \mathsf{i}E;\ K\rangle \qquad (\textsc{Close})$$

$$\langle \mathsf{i}\lambda x.e;\ E;\ \mathsf{i}E;\ K\rangle \mapsto \langle \mathtt{iclos}\,(x, e, E, ;)\ E;\ \mathsf{i}E;\ K\rangle \qquad (\textsc{iClose})$$

$$\langle e\ e';\ E;\ \mathsf{i}E;\ K\rangle \mapsto \langle e;\ E;\ \mathsf{i}E;\ (\mathtt{app}\,(\Box, e', E)\,, K)\rangle \qquad (\textsc{PushApp})$$

$$\langle e@e';\ E;\ \mathsf{i}E;\ K\rangle \mapsto \langle e;\ E;\ \mathsf{i}E;\ (\mathtt{iapp}\,(\Box, e', E, \mathsf{i}E)\,, K)\rangle \quad (\textsc{PushiApp})$$

$$\langle w;\ E;\ \mathsf{i}E;\ \mathtt{app}\,(\Box, e', E')\,, K\rangle \mapsto \langle e';\ E';\ \mathsf{i}E;\ (w\ \Box, K)\rangle \quad (\textsc{EvalArg})$$

$$\langle w;\ E;\ \mathsf{i}E;\ (\mathtt{iapp}\,(\Box, e', E', \mathsf{i}E')\,, K)\rangle \mapsto \langle e';\ E';\ \mathsf{i}E';\ (w@\Box, K)\rangle \quad (\textsc{EvaliArg})$$

$$\langle w;\ E;\ \mathsf{i}E;\ (\mathtt{clos}\,(\lambda x.e, E')\,, K)\rangle \mapsto \langle e;\ ((x = w)\,, E')\,;\ \mathsf{i}E;\ K\rangle \qquad (\textsc{App})$$

$$\langle w;\ E;\ \mathsf{i}E;\ (\mathtt{iclos}\,(x, e, E', \mathsf{i}E')\,, K)\rangle \mapsto \langle e;\ E';\ ((x = w)\,, \mathsf{i}E')\,;\ K\rangle \quad (\textsc{iApp})$$

$$\langle a;\ []\,;\ []\,;\ []\rangle \mapsto \text{Result: } a \qquad (\textsc{ResltAtom})$$

$$\langle \mathtt{clos}\,(\lambda x.e, E)\,;\ []\,;\ []\,;\ []\rangle \mapsto \text{Result: } \lambda x.e \qquad (\textsc{Result}\lambda)$$

$$\langle \mathtt{iclos}\,(x, e, E, ;)\ []\,;\ []\,;\ []\rangle \mapsto \text{Result: } \mathsf{i}\lambda x.e \qquad (\textsc{Resulti}\lambda)$$

3 Type System

$$\frac{}{\Gamma \mid \Delta \vdash a : A} \qquad (\textsc{Atom})$$

$$\Delta + \!\!+ \Delta' = \Delta''$$
$$\text{assumes } \forall x \in \Delta, \forall y \in \Delta'.\, (x \notin \Delta' \wedge (x \in \Delta'' \wedge y \in \Delta''))$$

$$\frac{(x = \tau)\,, \Gamma \mid [] \vdash e : \tau'}{\Gamma \mid \Delta \vdash \lambda x.e : \tau \to \tau'} \qquad (\lambda)$$

$$\frac{\Gamma \mid (x = \tau)\,, \Delta \vdash e : \tau'}{\Gamma \mid \Delta \vdash \mathsf{i}\lambda x.e : \tau \multimap \tau'} \qquad (\mathsf{i}\lambda)$$

$$\frac{\Gamma \mid [] \vdash e : \tau \to \tau' \qquad \Gamma \mid [] \vdash e' : \tau}{\Gamma \mid \Delta \vdash e\ e' : \tau'} \qquad (\textsc{App})$$

$$\frac{\Gamma \mid \Delta \vdash e : \tau \multimap \tau' \qquad \Gamma \mid \Delta' \vdash e' : \tau}{\Gamma \mid \Delta + \!\!+ \Delta' \vdash e@e' : \tau'} \qquad (\mathsf{i}\textsc{App})$$

$$\frac{(x = \tau) \in \Gamma}{\Gamma \mid \Delta \vdash x : \tau} \qquad (\textsc{Var})$$

$$\frac{(x = \tau) \in \Delta}{\Gamma \mid \Delta \vdash x : \tau} \qquad (\mathsf{i}\textsc{Var})$$

$$\frac{\Gamma \mid [] \vdash e : \tau \qquad (x = \tau)\,, \Gamma \mid \Delta \vdash e' : \tau'}{\Gamma \mid \Delta \vdash \texttt{let}\ x = e\ \texttt{in}\ e' : \tau'} \qquad (\textsc{Let})$$

$$\frac{\Gamma \mid \Delta \vdash e : \tau \qquad (x = \tau)\,, \Gamma \mid \Delta' \vdash e' : \tau'}{\Gamma \mid \Delta + \!\!+ \Delta' \vdash \texttt{ilet}\ x = e\ \texttt{in}\ e' : \tau'} \qquad (\mathsf{i}\textsc{Let})$$