1 Krivine abstract machine

1.1 Syntax

Unary numbers

%datatype *unary*
%name *unary* n

$$
\begin{array}{lll}
n & ::= & \\
 & | & 0 \\
 & | & n+1
\end{array}
$$

Term

%datatype *term*
%name *term* t

$$
\begin{array}{lll}
t & ::= & \\
 & | & n \\
 & | & t_1\, t_2 \\
 & | & \lambda t
\end{array}
$$

Closure and environment

%datatype *clos*
%name *clos* c

%datatype *env*
%name *env* $\mathcal{E}$

$$
\begin{array}{lll}
c & ::= & \\
 & | & (t, \mathcal{E})
\end{array}
$$

$$
\begin{array}{lll}
\mathcal{E} & ::= & \\
 & | & [] \\
 & | & (\mathcal{E}, c)
\end{array}
$$

Stack

%datatype *stack*
%name *stack* $\mathcal{S}$

$$
\begin{array}{lll}
\mathcal{S} & ::= & \\
 & | & [] \\
 & | & c :: \mathcal{S}
\end{array}
$$

State

%datatype *state*
%name *state* σ

$$
\begin{array}{lll}
\sigma & ::= & \\
 & | & \langle t, \mathcal{E}, \mathcal{S} \rangle
\end{array}
$$

1.2 Judgments

$\mathcal{E}(n) = c$: **type**.

$\sigma_1 \rightarrow \sigma_2 \quad : \quad \mathbf{type}.$

1.3 Fetch

$$\frac{}{(\mathcal{E}, c)(0) = c}\,[\mathrm{fetch}_1]$$

$$\frac{\mathcal{E}(n) = c}{(\mathcal{E}, c')(n+1) = c}\,[\mathrm{fetch}_2]$$

%mode $+\mathcal{E}(+n) = -c$
%worlds $()\quad \mathcal{E}(n) = c$
%terminates $\mathcal{E}\quad \mathcal{E}(n) = c$
%unique $+\mathcal{E}(+n) = -1c$

1.4 Evaluation

$$\frac{\mathcal{E}(n) = (t, \mathcal{E}')}{\langle n, \mathcal{E}, \mathcal{S}\rangle \rightarrow \langle t, \mathcal{E}', \mathcal{S}\rangle}\,[\mathrm{step\_var}]$$

$$\frac{}{\langle (t_1\, t_2), \mathcal{E}, \mathcal{S}\rangle \rightarrow \langle t_1, \mathcal{E}, (t_2, \mathcal{E}) :: \mathcal{S}\rangle}\,[\mathrm{step\_app}]$$

$$\frac{}{\langle \lambda t, \mathcal{E}, c :: \mathcal{S}\rangle \rightarrow \langle t, (\mathcal{E}, c), \mathcal{S}\rangle}\,[\mathrm{step\_abs}]$$

%mode $+\sigma_1 \rightarrow -\sigma_2$
%worlds $()\quad \sigma_1 \rightarrow \sigma_2$
%unique $+\sigma_1 \rightarrow -1\sigma_2$