1 Concrete Semantics of Scheme CESKt*

Syntax Domains:

$$
\begin{aligned}
e \in \mathsf{Exp} ::= {} & æ \\
& \mid (\texttt{if}\ e\ e\ e) \\
& \mid (\texttt{let}\ ([x\ e]\ ...)\ e) \\
& \mid (\texttt{call/cc}\ e) \\
& \mid (\texttt{set!}\ x\ e) \\
& \mid (\texttt{prim}\ op\ e\ ...) \\
& \mid (\texttt{apply-prim}\ op\ e) \\
& \mid (\texttt{apply}\ e\ e) \\
& \mid (e\ e\ ...) \\
æ \in \mathsf{AExp} ::= {} & x \mid lam \mid \mathbb{Z} \mid \texttt{\#t} \mid \texttt{\#f} \\
& \mid (\texttt{quote}\ e) \\
lam \in \mathsf{Lam} ::= {} & (\lambda\ (x...)\ e) \mid (\lambda\ x\ e) \\
x \in \mathsf{Var} & \quad \text{A set of identifiers} \\
op \in \mathsf{Prim} & \quad \text{A set of primitives}
\end{aligned}
$$

Atomic Evaluation:

$$
\begin{aligned}
\mathcal{A} : \Sigma_E \times \sigma & \rightharpoonup Val \\
\mathcal{A}(\langle lam, \rho, \_, \_\rangle, \_) & \triangleq (lam, \rho) \\
\mathcal{A}(\langle (\texttt{quote}\ e), \_, \_, \_\rangle, \_) & \triangleq \mathbf{quote}(e) \\
\mathcal{A}(\langle x, \rho, \_, \_\rangle, \sigma) & \triangleq \sigma(\rho(x)) \\
\mathcal{A}(\langle æ, \_, \_, \_\rangle, \_) & \triangleq æ
\end{aligned}
$$

Tick/Alloc:

$$
\begin{aligned}
tick : \Sigma \times \mathbb{N} & \to Time \\
tick(\langle \_, \_, \_, t\rangle, n) & \triangleq (t + n) \\
alloc : \Sigma \times \mathbb{N} & \triangleq Addr \\
alloc(\langle \_, \_, \_, t\rangle, n) & \triangleq (t + n)
\end{aligned}
$$

Injection:

$$
\begin{aligned}
\mathcal{I} : \mathsf{Exp} & \to \Sigma \\
\mathcal{I}(e) & \triangleq (e, \varnothing, 0, 1)
\end{aligned}
$$

Initial σ state $\triangleq \{0 : \mathbf{mt}\}$

Transition:

Collecting Semantics:

Semantic Domains:

$$
\begin{aligned}
\varsigma \in \Sigma & \triangleq E\langle Eval\rangle + A\langle Apply\rangle \\
Eval & \triangleq \mathsf{Exp} \times Env \\
& \quad \times Addr \times Time \\
Apply & \triangleq Val \times Env \\
& \quad \times Addr \times Time \\
\rho \in Env & \triangleq \mathsf{Var} \rightharpoonup Addr \\
\sigma \in Store & \triangleq Addr \rightharpoonup Val \\
v \in Val & \triangleq Clo + Kont + \mathbb{Z} \\
& \quad + \{\texttt{\#t}, \texttt{\#f}, Null, Void\} \\
& \quad + \{\mathbf{quote}(e), \mathbf{cons}(v, v)\} \\
clo \in Clo & \triangleq \mathsf{Lam} \times Env \\
a, b, c \in Addr & \triangleq \mathbb{N} \\
t, u \in Time & \triangleq \mathbb{N} \\
\kappa \in Kont & ::= \mathbf{mt} \\
& \mid \mathbf{ifk}(e, e, \rho, a) \\
& \mid \mathbf{callcck}(\rho, a) \\
& \mid \mathbf{setk}(x, \rho, a) \\
& \mid \mathbf{appappk}(val?, e, \rho, a) \\
& \mid \mathbf{appk}(done, todo, \rho, a) \\
& \mid \mathbf{appprimk}(op, \rho, a) \\
& \mid \mathbf{primk}(op, done, todo, \\
& \qquad \rho, a) \\
& \mid \mathbf{letk}(vars, done, todo, \\
& \qquad e, \rho, a) \\
done & \triangleq Val^* \\
todo & \triangleq \mathsf{Exp}^* \\
vars & \triangleq \mathsf{Var}^*
\end{aligned}
$$

Eval Rules

Rules for when the control is an expression

$$E\langle æ, \rho, a, t\rangle \rightsquigarrow A\langle v, \rho, a, t\rangle$$
$$\text{where } v \triangleq \mathcal{A}(\varsigma, \sigma)$$

$$E\langle(\texttt{if}\ e_c\ e_t\ e_f), \rho, a, t\rangle \rightsquigarrow E\langle e_c, \rho, b, u\rangle$$
$$\text{where } b \triangleq alloc(\varsigma, 0)$$
$$u \triangleq tick(\varsigma, 1)$$
$$\sigma[b] \triangleq \mathbf{ifk}(e_t, e_f, \rho, a)$$

$$E\langle(\texttt{let}\ ()\ e), \rho, a, t\rangle \rightsquigarrow E\langle e, \rho, a, t\rangle$$

$$E\langle(\texttt{let}\ ([x_0\ e_0]\ [x_s\ e_s]\ ...)\ e_b), \rho, a, t\rangle$$
$$\rightsquigarrow E\langle e_0, \rho, b, u\rangle$$
$$\text{where } b \triangleq alloc(\varsigma, 0)$$
$$u \triangleq tick(\varsigma, 1)$$
$$\sigma[b] \triangleq \mathbf{letk}(x_0 :: x_s, [\,], e_s, e_b, \rho, a)$$

$$E\langle(\texttt{call/cc}\ e), \rho, a, t\rangle \rightsquigarrow E\langle e, \rho, b, u\rangle$$
$$\text{where } b \triangleq alloc(\varsigma, 0)$$
$$u \triangleq tick(\varsigma, 1)$$
$$\sigma[b] \triangleq \mathbf{callcck}(\rho, a)$$

$$E\langle(\texttt{set!}\ x\ e), \rho, a, t\rangle \rightsquigarrow E\langle e, \rho, b, u\rangle$$
$$\text{where } b \triangleq alloc(\varsigma, 0)$$
$$u \triangleq tick(\varsigma, 1)$$
$$\sigma[b] \triangleq \mathbf{setk}(x, a)$$

$$E\langle(\texttt{prim}\ op), \rho, a, t\rangle \rightsquigarrow A\langle v, \rho, a, t\rangle$$
$$\text{where } v = op \text{ applied to 0 arguments}$$

$$E\langle(\texttt{prim}\ op\ e_0\ e_s\ ...), \rho, a, t\rangle$$
$$\rightsquigarrow E\langle e_0, \rho, b, u\rangle$$
$$\text{where } b \triangleq alloc(\varsigma, 0)$$
$$u \triangleq tick(\varsigma, 1)$$
$$\sigma[b] \triangleq \mathbf{primk}(op, [\,], e_s, \rho, a)$$

$$E\langle(\texttt{apply-prim}\ op\ e), \rho, a, t\rangle$$
$$\rightsquigarrow E\langle e, \rho, b, u\rangle$$
$$\text{where } b \triangleq alloc(\varsigma, 0)$$
$$u \triangleq tick(\varsigma, 1)$$
$$\sigma[b] \triangleq \mathbf{appprimk}(op, a)$$

$$E\langle(\texttt{apply}\ e_f\ e_x), \rho, a, t\rangle$$
$$\rightsquigarrow E\langle e_f, \rho, b, u\rangle$$
$$\text{where } b \triangleq alloc(\varsigma, 0)$$
$$u \triangleq tick(\varsigma, 1)$$
$$\sigma[b] \triangleq \mathbf{appappk}(\varnothing, e_x, \rho, a)$$

$$E\langle(e_f\ e_s\ ...), \rho, a, t\rangle \rightsquigarrow E\langle e_f, \rho, b, u\rangle$$
$$\text{where } b \triangleq alloc(\varsigma, 0)$$
$$u \triangleq tick(\varsigma, 1)$$
$$\sigma[b] \triangleq \mathbf{appk}([\,], e_s, \rho, a)$$

Apply Rules

Rules for when the control is a value

$$\varsigma = A\langle v, \rho, a, t\rangle$$

$$\kappa \triangleq \sigma(a)$$

Proceed by matching on κ

$$\mathbf{mt} \rightsquigarrow \varsigma$$

$$\mathbf{ifk}(e_t, e_f, \rho_\kappa, c) \rightsquigarrow E\langle e_f, \rho_\kappa, c, t\rangle$$
where $v = \texttt{\#f}$

$$\mathbf{ifk}(e_t, e_f, \rho_\kappa, c) \rightsquigarrow E\langle e_t, \rho_\kappa, c, t\rangle$$
where $v \neq \texttt{\#f}$

$$\mathbf{letk}(vars, done, [\,], e_b, \rho_\kappa, c) \rightsquigarrow E\langle e_b, \rho'_\kappa, c, u\rangle$$
where $b_i \triangleq alloc(\varsigma, i)$

$u \triangleq tick(\varsigma, n)$

$\rho'_\kappa \triangleq \rho_\kappa[vars_0 \mapsto b_0 \; ... \; vars_{n-1} \mapsto b_{n-1}]$

$done' \triangleq done \mathbin{+\!\!+} [v]$

$\sigma[b_0 \; ... \; b_{n-1}] \triangleq done'_0 \; ... \; done'_{n-1}$

$$\mathbf{letk}(vars, done, e_h :: e_t, e_b, \rho_\kappa, c) \rightsquigarrow E\langle e_h, \rho_\kappa, b, u\rangle$$
where $b \triangleq alloc(\varsigma, 0)$

$u \triangleq tick(\varsigma, 1)$

$\sigma[b] \triangleq \mathbf{letk}(vars, done \mathbin{+\!\!+} [v], e_t, e_b, \rho_\kappa, c)$

$$\mathbf{callcck}(\_, c) \rightsquigarrow E\langle e, \rho'_\lambda, c, t\rangle$$
where $v = ((\lambda \; (x) \; e), \rho_\lambda)$

$\rho'_\lambda \triangleq \rho_\lambda[x \mapsto c]$

$$\mathbf{callcck}(\rho_\kappa, \_) \rightsquigarrow A\langle \kappa, \rho_\kappa, b, u\rangle$$
where $v = \kappa'$

$b \triangleq alloc(\varsigma, 0)$

$u \triangleq tick(\varsigma, 1)$

$\sigma[b] \triangleq \kappa'$

$$\mathbf{setk}(x, \rho_\kappa, c) \rightsquigarrow A\langle Void, \rho_\kappa, c, t\rangle$$
where $\sigma[\rho_\kappa(x)] \triangleq v$

$$\mathbf{appprimk}(op, \rho_\kappa, c) \rightsquigarrow A\langle v', \rho_\kappa, c, t\rangle$$
where $v' \triangleq op$ applied to v

$$\mathbf{primk}(op, done, [\,], \rho_\kappa, c) \rightsquigarrow A\langle v', \rho_\kappa, c, t\rangle$$
where $v' \triangleq op$ applied to $(done \mathbin{+\!\!+} [v])$

$$\mathbf{primk}(op, done, e_h :: e_t, \rho_\kappa, c) \rightsquigarrow E\langle e_h, \rho_\kappa, b, u\rangle$$
where $b \triangleq alloc(\varsigma, 0)$

$u \triangleq tick(\varsigma, 1)$

$\sigma[b] \triangleq \mathbf{primk}(op, done \mathbin{+\!\!+} [v], e_t, \rho_\kappa, c)$

More Apply Rules

Rules for when the control is a value

$$\textbf{appappk}(\varnothing, e, \rho_\kappa, c) \rightsquigarrow E\langle e, \rho_\kappa, b, u\rangle$$
$$\begin{aligned} \text{where } b &\triangleq \mathit{alloc}(\varsigma, 0) \\ u &\triangleq \mathit{tick}(\varsigma, 1) \\ \sigma[b] &\triangleq \textbf{appappk}(v, e, \rho_\kappa, c) \end{aligned}$$

$$\textbf{appappk}(v_f, \_, \_, c) \rightsquigarrow E\langle e_b, \rho'_\lambda, c, u\rangle$$
$$\begin{aligned} \text{where } v_f &= ((\lambda\ (x_s ...)\ e_b), \rho_\lambda) \\ b_i &\triangleq \mathit{alloc}(\varsigma, i) \\ u &\triangleq \mathit{tick}(\varsigma, n) \\ \rho'_\lambda &\triangleq \rho_\lambda[x_0 \mapsto b_0\ ... \\ &\qquad x_{n-1} \mapsto b_{n-1}] \\ \sigma[b_0\ ...\ b_{n-1}] &\triangleq v_0\ ...\ v_{n-1} \end{aligned}$$

$$\textbf{appappk}(v_f, \_, \_, c) \rightsquigarrow E\langle e_b, \rho'_\lambda, c, u\rangle$$
$$\begin{aligned} \text{where } v_f &= ((\lambda\ x\ e_b), \rho_\lambda) \\ b &\triangleq \mathit{alloc}(\varsigma, 0) \\ u &\triangleq \mathit{tick}(\varsigma, 1) \\ \rho'_\lambda &\triangleq \rho_\lambda[x \mapsto b] \\ \sigma[b] &\triangleq v \end{aligned}$$

$$\textbf{appk}(\mathit{done}, [\,], \_, c) \rightsquigarrow E\langle e_b, \rho'_\lambda, c, u\rangle$$
$$\begin{aligned} \text{where } \mathit{done} &= ((\lambda\ (x_s ...)\ e_b), \rho_\lambda) :: v_s \\ b_i &\triangleq \mathit{alloc}(\varsigma, i) \\ u &\triangleq \mathit{tick}(\varsigma, n) \\ \rho'_\lambda &\triangleq \rho_\lambda[x_0 \mapsto b_0\ ... \\ &\qquad x_{n-1} \mapsto b_{n-1}] \\ v'_s &\triangleq v_s +\!\!+ [v] \\ \sigma[b_0] &\triangleq v'_0\ ...\ b_{n-1} \mapsto v'_{n-1} \end{aligned}$$

$$\textbf{appk}(\mathit{done}, [\,], \_, c) \rightsquigarrow E\langle e_b, \rho'_\lambda, c, u\rangle$$
$$\begin{aligned} \text{where } \mathit{done} &= ((\lambda\ x\ e_b), \rho_\lambda) :: v_s \\ b &\triangleq \mathit{alloc}(\varsigma, 0) \\ u &\triangleq \mathit{tick}(\varsigma, 1) \\ \rho'_\lambda &\triangleq \rho_\lambda[x \mapsto b] \\ v'_s &\triangleq (v_s +\!\!+ [v]) \\ \sigma[b] &\triangleq v'_s \end{aligned}$$

$$\textbf{appk}([\kappa_\lambda], [\,], \rho_\kappa, c) \rightsquigarrow A\langle v, \rho_\kappa, b, u\rangle$$
$$\begin{aligned} \text{where } b &\triangleq \mathit{alloc}(\varsigma, 0) \\ u &\triangleq \mathit{tick}(\varsigma, 1) \\ \sigma[b] &\triangleq \kappa_\lambda \end{aligned}$$

$$\begin{aligned} &\textbf{appk}(\mathit{done}, e_h :: e_t, \rho_\kappa, c) \\ &\quad \rightsquigarrow E\langle e_h, \rho_\kappa, b, u\rangle \end{aligned}$$
$$\begin{aligned} \text{where } b &\triangleq \mathit{alloc}(\varsigma, 0) \\ u &\triangleq \mathit{tick}(\varsigma, 1) \\ \sigma[b] &\triangleq \textbf{appk}(\mathit{done} +\!\!+ [v], \\ &\qquad e_t, \rho_\kappa, c) \end{aligned}$$

2 Abstract Semantics of Scheme CESKt*

Tick/Alloc:

$$\widehat{tick} : \hat{\Sigma} \times Kont \rightarrow Time$$

$$\widehat{tick}(\hat{\varsigma}, \kappa) \triangleq 0$$

$$\widehat{alloc} : \hat{\Sigma} \times Kont \rightarrow Addr$$

$$\widehat{alloc}(\hat{\varsigma}, \kappa) \triangleq 0$$

Eval Rules

$$E\langle(\texttt{if}\ e_c\ e_t\ e_f), \rho, a, t\rangle \rightsquigarrow E\langle e_c, \rho, b, u\rangle$$

$$\text{where } b \triangleq alloc(\varsigma, 0)$$

$$u \triangleq tick(\varsigma, 1)$$

$$\sigma[b] \sqcup \mathbf{ifk}(e_t, e_f, \rho, a)$$

$$E\langle(\texttt{let}\ ()\ e), \rho, a, t\rangle \rightsquigarrow E\langle e, \rho, a, t\rangle$$