# 1 Concrete Semantics of Scheme CESK*

**Syntax Domains:**

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

$x \in \mathsf{Var}$ A set of identifiers

$op \in \mathsf{Prim}$ A set of primitives

**Atomic Evaluation:**

$$\mathcal{A} : \Sigma_E \times \sigma \rightharpoonup Val$$

$$\mathcal{A}(\langle lam, \rho, \_, \_\rangle, \_) \triangleq (lam, \rho)$$

$$\mathcal{A}(\langle (\texttt{quote}\ e), \_, \_, \_\rangle, \_) \triangleq \mathbf{quote}(e)$$

$$\mathcal{A}(\langle x, \rho, \_, \_\rangle, \sigma) \triangleq \sigma(\rho(x))$$

$$\mathcal{A}(\langle æ, \_, \_, \_\rangle, \_) \triangleq æ$$

**Tick/Alloc:**

$$tick : \Sigma \times \mathbb{N} \to Time$$

$$tick(\langle \_, \_, \_, t\rangle, n) \triangleq (t+n)$$

$$alloc : \Sigma \times \mathbb{N} \triangleq Addr$$

$$alloc(\langle \_, \_, \_, t\rangle, n) \triangleq (t+n)$$

**Injection:**

$$\mathcal{I} : \mathsf{Exp} \to \Sigma$$

$$\mathcal{I}(e) \triangleq (e, \varnothing, 0, 1)$$

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

**Transition:**

**Collecting Semantics:**

**Semantic Domains:**

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

## Eval Rules

Rules for when the control is an expression

$$E\langle æ, \rho, a, t\rangle \rightsquigarrow A\langle v, \rho, a, u\rangle$$
$$\text{where } u = \mathit{tick}(st, 1)$$
$$v = \mathcal{A}(\varsigma, \sigma)$$

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

$$E\langle(\texttt{let}\ ()\ e), \rho, a, t\rangle \rightsquigarrow E\langle e, \rho, a, u\rangle$$
$$\text{where } u = \mathit{tick}(\varsigma, 1)$$

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

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

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

$$E\langle(\texttt{prim}\ \mathit{op}), \rho, a, t\rangle \rightsquigarrow A\langle v, \rho, a, u\rangle$$
$$\text{where } v = \mathit{op} \text{ applied to 0 arguments}$$
$$u = \mathit{tick}(\varsigma, 1)$$

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

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

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

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

**Apply Rules**

Rules for when the control is a value

$$A\langle v, \rho, a, t\rangle \rightsquigarrow E\langle e_f, \rho, c, u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{ifk}(e_t, e_f, \rho, c)$$
$$u = tick(\varsigma, 1)$$
$$v = \texttt{\#f}$$

$$A\langle v, \rho, a, t\rangle \rightsquigarrow E\langle e_t, \rho, c, u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{ifk}(e_t, e_f, \rho, c)$$
$$u = tick(\varsigma, 1)$$
$$v \neq \texttt{\#f}$$

$$A\langle v, \rho, a, t\rangle \rightsquigarrow E\langle e_b, \rho'_\kappa, c, u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{letk}(vars, done, [\,], e_b, \rho_\kappa, c)$$
$$\rho'_\kappa = \rho_\kappa[vars_0 \mapsto b_0 \ldots vars_n \mapsto b_n]$$
$$\sigma[b_0 \mapsto done_i \ldots b_n \mapsto done_n]$$
$$b_i = alloc(\varsigma, i)$$
$$u = tick(\varsigma, n)$$

$$A\langle v, \rho, a, t\rangle \rightsquigarrow E\langle e_h, \rho_\kappa, b, u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{letk}(vars, done, e_h :: e_t, e_b, \rho_\kappa, c)$$
$$\sigma[b \mapsto \mathbf{letk}(vars, done +\!\!+ [v], e_t, e_b, \rho_\kappa, c)]$$
$$b = alloc(\varsigma, 0)$$
$$u = tick(\varsigma, 1)$$

$$A\langle v, \rho, a, t\rangle \rightsquigarrow E\langle e, \rho', c, u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{callcck}(c)$$
$$v = ((\lambda\ (x)\ e), \rho_\lambda)$$
$$\rho' = \rho_\lambda[x \mapsto c]$$
$$u = tick(\varsigma, 1)$$

$$A\langle v, \rho, a, t\rangle \rightsquigarrow E\langle e, \rho', c, u\rangle$$
where $\kappa \triangleq \sigma(a)$ **TODO**
$$\kappa = \mathbf{callcck}(c)$$
$$v = \kappa'$$
$$\rho' = \rho_\lambda[x \mapsto c]$$
$$u = tick(\varsigma, 1)$$

$$A\langle v, \rho, a, t\rangle \rightsquigarrow A\langle Void, \rho_\kappa, c, u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{setk}(x, \rho_\kappa, c)$$
$$\sigma[\rho_\kappa(x) \mapsto v]$$
$$u = tick(\varsigma, 1)$$

$$A\langle v, \rho, a, t\rangle \rightsquigarrow E\langle e, \rho_\kappa, b, u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{appappk}(\varnothing, e, \rho_\kappa, c)$$
$$\sigma[b \mapsto \mathbf{appappk}(v, e, \rho_\kappa, c)]$$
$$b = alloc(\varsigma, 0)$$
$$u = tick(\varsigma, 1)$$

**More Apply Rules**

Rules for when the control is a value

$$A\langle v,\rho,a,t\rangle \rightsquigarrow E\langle e_b,\rho'_\lambda,c,u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{appappk}(v_f, \_, \rho_\kappa, c)$$
$$v_f = ((\lambda\ (x_s...)\ e_b), \rho_\lambda)$$
$$\rho'_\lambda = \rho_\lambda[x_0 \mapsto b_0\ ...\ x_n \mapsto b_n]$$
$$\sigma[b_0 \mapsto v_0\ ...\ b_n \mapsto v_n]$$
$$b_i = alloc(\varsigma, i)$$
$$u = tick(\varsigma, n)$$

$$A\langle v,\rho,a,t\rangle \rightsquigarrow E\langle e_b,\rho'_\lambda,c,u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{appappk}(v_f, \_, \rho_\kappa, c)$$
$$v_f = ((\lambda\ x\ e_b), \rho_\lambda)$$
$$\rho'_\lambda = \rho_\lambda[x \mapsto b]$$
$$\sigma[b \mapsto v]$$
$$b = alloc(\varsigma, 0)$$
$$u = tick(\varsigma, 1)$$

$$A\langle v,\rho,a,t\rangle \rightsquigarrow E\langle e_h,\rho_\kappa,b,u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{appk}(done, e_h :: e_t, \rho_\kappa, c)$$
$$\sigma[b \mapsto \mathbf{appk}(done +\!\!+ [v], e_t, \rho_\kappa, c)]$$
$$b = alloc(\varsigma, 0)$$
$$u = tick(\varsigma, 1)$$

$$A\langle v,\rho,a,t\rangle \rightsquigarrow E\langle e_b,\rho'_\lambda,c,u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{appk}(done, [\,], \rho_\kappa, c)$$
$$done = ((\lambda\ (x_s...)\ e_b), \rho_\lambda) :: v_s$$
$$\rho'_\lambda = \rho_\lambda[x_0 \mapsto b_0\ ...\ x_n \mapsto b_n]$$
$$v'_s = v_s +\!\!+ [v]$$
$$\sigma[b_0 \mapsto v'_0\ ...\ b_n \mapsto v'_n]$$
$$b_i = alloc(\varsigma, i)$$
$$u = tick(\varsigma, n)$$

$$A\langle v,\rho,a,t\rangle \rightsquigarrow E\langle e_b,\rho'_\lambda,c,u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{appk}(done, [\,], \rho_\kappa, c)$$
$$done = ((\lambda\ x\ e_b), \rho_\lambda) :: v_s$$
$$\rho'_\lambda = \rho_\lambda[x \mapsto b]$$
$$v' = (done +\!\!+ [v]) \text{ as a cons-list}$$
$$\sigma[b \mapsto v']$$
$$b = alloc(\varsigma, 0)$$
$$u = tick(\varsigma, 1)$$

$$A\langle v,\rho,a,t\rangle \rightsquigarrow A\langle v,\rho_\kappa,b,u\rangle$$
where $\kappa \triangleq \sigma(a)$
$$\kappa = \mathbf{appk}([\kappa_\lambda], [\,], \rho_\kappa, c)$$
$$\sigma[b \mapsto \kappa_\lambda]$$
$$b = alloc(\varsigma, 0)$$
$$u = tick(\varsigma, 1)$$

**More Apply Rules**

Rules for when the control is a value

$$\begin{aligned}
&A\langle v,\rho,a,t\rangle \rightsquigarrow A\langle v',\rho',c,u\rangle\\
&\text{where } \kappa \triangleq \sigma(a)\\
&\qquad \kappa = \mathbf{appprimk}(op,\rho',c)\\
&\qquad v' = op \text{ applied to } v\\
&\qquad u = tick(\varsigma,1)
\end{aligned}$$

$$\begin{aligned}
&A\langle v,\rho,a,t\rangle \rightsquigarrow E\langle e_h,\rho',b,u\rangle\\
&\text{where } \kappa \triangleq \sigma(a)\\
&\qquad \kappa = \mathbf{primk}(op, done, e_h :: e_t,\\
&\qquad\qquad \rho', c)\\
&\qquad \sigma[b \mapsto \mathbf{primk}(op, done +\!\!+ [v],\\
&\qquad\qquad e_t, \rho', c)]\\
&\qquad b = alloc(\varsigma,0)\\
&\qquad u = tick(\varsigma,1)
\end{aligned}$$

$$\begin{aligned}
&A\langle v,\rho,a,t\rangle \rightsquigarrow A\langle v',\rho',c,u\rangle\\
&\text{where } \kappa \triangleq \sigma(a)\\
&\qquad \kappa = \mathbf{primk}(op, done, [\,], \rho', c)\\
&\qquad v' = op \text{ applied to } (done +\!\!+ [v])\\
&\qquad u = tick(\varsigma,1)
\end{aligned}$$

## 2 Abstract Semantics of Scheme CESK*