

1 Concrete Semantics of Scheme CESK*

Syntax Domains:

$e \in \text{Exp} ::= \text{\ae}$
 $\quad | (\text{if } e \ e \ e)$
 $\quad | (\text{let } ([x \ e] \ \dots) \ e)$
 $\quad | (\text{call/cc } e)$
 $\quad | (\text{set! } x \ e)$
 $\quad | (\text{prim } op \ e \ \dots)$
 $\quad | (\text{apply-prim } op \ e)$
 $\quad | (\text{apply } e \ e)$
 $\quad | (e \ e \ \dots)$

$\text{\ae} \in \text{AExp} ::= x \mid lam \mid \mathbb{Z} \mid \#t \mid \#f$
 $\quad | (\text{quote } e)$

$lam \in \text{Lam} ::= (\lambda (x \dots) e) \mid (\lambda x \ e)$

$x \in \text{Var} \quad \text{A set of identifiers}$

$op \in \text{Prim} \quad \text{A set of primitives}$

Atomic Evaluation:

$\mathcal{A} : \Sigma_E \times \sigma \rightarrow \text{Val}$
 $\mathcal{A}(\langle lam, \rho, -, - \rangle, -) \triangleq (lam, \rho)$
 $\mathcal{A}(\langle (\text{quote } e), -, -, - \rangle, -) \triangleq \text{quote}(e)$
 $\mathcal{A}(\langle x, \rho, -, - \rangle, \sigma) \triangleq \sigma(\rho(x))$
 $\mathcal{A}(\langle \text{\ae}, -, -, - \rangle, -) \triangleq \text{\ae}$

Tick/Alloc:

$tick : \Sigma \times \mathbb{N} \rightarrow \text{Time}$
 $tick(\langle -, -, -, t \rangle, n) \triangleq (t + n)$
 $alloc : \Sigma \times \mathbb{N} \triangleq \text{Addr}$
 $alloc(\langle -, -, -, t \rangle, n) \triangleq (t + n)$

Injection:

$\mathcal{I} : \text{Exp} \rightarrow \Sigma$
 $\mathcal{I}(e) \triangleq (e, \emptyset, 0, 1)$
 $\text{Initial } \sigma \text{ state} \triangleq \{0 : \text{mt}\}$

Transition:

Collecting Semantics:

Semantic Domains:

$\varsigma \in \Sigma \triangleq E \langle \text{Eval} \rangle + A \langle \text{Apply} \rangle$

$\text{Eval} \triangleq \text{Exp} \times \text{Env}$
 $\quad \times \text{Addr} \times \text{Time}$

$\text{Apply} \triangleq \text{Val} \times \text{Env}$
 $\quad \times \text{Addr} \times \text{Time}$

$\rho \in \text{Env} \triangleq \text{Var} \rightarrow \text{Addr}$

$\sigma \in \text{Store} \triangleq \text{Addr} \rightarrow \text{Val}$

$v \in \text{Val} \triangleq \text{Clo} + \text{Kont} + \mathbb{Z}$
 $\quad + \{\#t, \#f, \text{Null}, \text{Void}\}$
 $\quad + \text{quote}(e) + \text{cons}(v, v)$

$clo \in \text{Clo} \triangleq \text{Lam} \times \text{Env}$

$a, b, c \in \text{Addr} \triangleq \mathbb{N}$

$t, u \in \text{Time} \triangleq \mathbb{N}$

$\kappa \in \text{Kont} ::= \text{mt}$

$\quad | \text{ifk}(e, e, \rho, a)$
 $\quad | \text{calleck}(a)$
 $\quad | \text{setk}(x, \rho, a)$
 $\quad | \text{appappk}(\text{val?}, e, \rho, a)$
 $\quad | \text{appk}(\text{done}, \text{todo}, \rho, a)$
 $\quad | \text{appprimk}(op, \rho, a)$
 $\quad | \text{primk}(op, \text{done}, \text{todo},$
 $\quad \quad \rho, a)$
 $\quad | \text{letk}(\text{vars}, \text{done}, \text{todo}$
 $\quad \quad e, \rho, a)$

$\text{done} \triangleq \text{Val}^*$

$\text{todo} \triangleq \text{Exp}^*$

$\text{vars} \triangleq \text{Var}^*$

Eval Rules

Rules for when the control is an expression

$$\begin{array}{l}
E\langle \mathbf{\lambda}, \rho, a, t \rangle \rightsquigarrow A\langle v, \rho, a, t \rangle \\
\text{where } v \triangleq \mathcal{A}(\varsigma, \sigma) \\
\\
E\langle (\mathbf{if } e_c e_t e_f), \rho, a, t \rangle \rightsquigarrow E\langle e_c, \rho, b, u \rangle \quad E\langle (\mathbf{prim } op), \rho, a, t \rangle \rightsquigarrow A\langle v, \rho, a, t \rangle \\
\text{where } b \triangleq \mathit{alloc}(\varsigma, 0) \quad \text{where } v = op \text{ applied to 0 arguments} \\
\quad u \triangleq \mathit{tick}(\varsigma, 1) \\
\quad \sigma[b \mapsto \mathbf{ifk}(e_t, e_f, \rho, a)] \quad E\langle (\mathbf{prim } op e_0 e_s \dots), \rho, a, t \rangle \\
\quad \rightsquigarrow E\langle e_0, \rho, b, u \rangle \\
\\
E\langle (\mathbf{let } () e), \rho, a, t \rangle \rightsquigarrow E\langle e, \rho, a, u \rangle \quad \text{where } b \triangleq \mathit{alloc}(\varsigma, 0) \\
\text{where } u \triangleq \mathit{tick}(\varsigma, 1) \quad u \triangleq \mathit{tick}(\varsigma, 1) \\
\quad \sigma[b \mapsto \mathbf{primk}(op, [], e_s, \rho, a)] \\
\\
E\langle (\mathbf{let } ([x_0 e_0] [x_s e_s] \dots) e_b), \rho, a, t \rangle \rightsquigarrow E\langle e_0, \rho, b, u \rangle \\
\text{where } b \triangleq \mathit{alloc}(\varsigma, 0) \quad E\langle (\mathbf{apply-prim } op e), \rho, a, t \rangle \\
\quad u \triangleq \mathit{tick}(\varsigma, 1) \quad \rightsquigarrow E\langle e, \rho, b, u \rangle \\
\quad \sigma[b \mapsto \mathbf{letk}(x_0 :: x_s, \quad \text{where } b \triangleq \mathit{alloc}(\varsigma, 0) \\
\quad \quad [], e_s, e_b, \rho, a)] \quad u \triangleq \mathit{tick}(\varsigma, 1) \\
\quad \sigma[b \mapsto \mathbf{appprimk}(op, a)] \\
\\
E\langle (\mathbf{call/cc } e), \rho, a, t \rangle \rightsquigarrow E\langle e, \rho, b, u \rangle \quad E\langle (\mathbf{apply } e_f e_x), \rho, a, t \rangle \\
\text{where } b \triangleq \mathit{alloc}(\varsigma, 0) \quad \rightsquigarrow E\langle e_f, \rho, b, u \rangle \\
\quad u \triangleq \mathit{tick}(\varsigma, 1) \quad \text{where } b \triangleq \mathit{alloc}(\varsigma, 0) \\
\quad \sigma[b \mapsto \mathbf{callcck}(a)] \quad u \triangleq \mathit{tick}(\varsigma, 1) \\
\quad \sigma[b \mapsto \mathbf{appappk}(\emptyset, e_x, \rho, a)] \\
\\
E\langle (\mathbf{set! } x e), \rho, a, t \rangle \rightsquigarrow E\langle e, \rho, b, u \rangle \quad E\langle (e_f e_s \dots), \rho, a, t \rangle \rightsquigarrow E\langle e_f, \rho, b, u \rangle \\
\text{where } b \triangleq \mathit{alloc}(\varsigma, 0) \quad \text{where } b \triangleq \mathit{alloc}(\varsigma, 0) \\
\quad u \triangleq \mathit{tick}(\varsigma, 1) \quad u \triangleq \mathit{tick}(\varsigma, 1) \\
\quad \sigma[b \mapsto \mathbf{setk}(x, a)] \quad \sigma[b \mapsto \mathbf{appk}([], e_s, \rho, a)]
\end{array}$$

Apply Rules

Rules for when the control is a value

$$\begin{aligned}\varsigma &= A\langle v, \rho, a, t \rangle \\ \kappa &\triangleq \sigma(a)\end{aligned}$$

Proceed by matching on κ

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

$$\begin{aligned}\mathbf{ifk}(e_t, e_f, \rho, c) &\rightsquigarrow E\langle e_f, \rho, c, t \rangle \\ &\text{where } v = \#f\end{aligned}$$

$$\begin{aligned}\mathbf{ifk}(e_t, e_f, \rho, c) &\rightsquigarrow E\langle e_t, \rho, c, t \rangle \\ &\text{where } v \neq \#f\end{aligned}$$

$$\begin{aligned}\mathbf{letk}(vars, done, [], e_b, \rho_\kappa, c) & \\ &\rightsquigarrow E\langle e_b, \rho'_\kappa, c, u \rangle\end{aligned}$$

$$\text{where } b_i \triangleq alloc(\varsigma, i)$$

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

$$done' \triangleq done \# [v]$$

$$\begin{aligned}\rho'_\kappa &\triangleq \rho_\kappa[vars_0 \mapsto b_0 \dots \\ &\quad vars_{n-1} \mapsto b_{n-1}]\end{aligned}$$

$$\sigma[b_0 \mapsto done'_0 \dots b_{n-1} \mapsto done'_{n-1}]$$

$$\begin{aligned}\mathbf{letk}(vars, done, e_h :: e_t, e_b, \rho_\kappa, c) & \\ &\rightsquigarrow E\langle e_h, \rho_\kappa, b, u \rangle\end{aligned}$$

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

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

$$\begin{aligned}\sigma[b \mapsto \mathbf{letk}(vars, done \# [v], \\ e_t, e_b, \rho_\kappa, c)]\end{aligned}$$

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

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

TODO WTF IS THIS CASE

$$\begin{aligned}\mathbf{callcck}(c) &\rightsquigarrow E\langle e, \rho', c, t \rangle \\ &\text{where } v = \kappa'\end{aligned}$$

$$\rho' \triangleq \rho[x \mapsto c]$$

$$\begin{aligned}\mathbf{setk}(x, \rho_\kappa, c) &\rightsquigarrow A\langle Void, \rho_\kappa, c, t \rangle \\ &\text{where } \sigma[\rho_\kappa(x) \mapsto v]\end{aligned}$$

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

$$\begin{aligned}\mathbf{primk}(op, done, [], \rho_\kappa, c) & \\ &\rightsquigarrow A\langle v', \rho_\kappa, c, t \rangle \\ &\text{where } v' \triangleq op \text{ applied to } (done \# [v])\end{aligned}$$

$$\begin{aligned}\mathbf{primk}(op, done, e_h :: e_t, \rho_\kappa, c) & \\ &\rightsquigarrow E\langle e_h, \rho_\kappa, b, u \rangle\end{aligned}$$

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

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

$$\begin{aligned}\sigma[b \mapsto \mathbf{primk}(op, done \# [v], \\ e_t, \rho_\kappa, c)]\end{aligned}$$

More Apply Rules

Rules for when the control is a value

$$\begin{array}{ll}
\mathbf{appappk}(\emptyset, e, \rho_\kappa, c) \rightsquigarrow E\langle e, \rho_\kappa, b, u \rangle & \mathbf{appk}(done, [], \rho_\kappa, c) \rightsquigarrow E\langle e_b, \rho'_\lambda, c, u \rangle \\
\text{where } b \triangleq alloc(\varsigma, 0) & \text{where } done = ((\lambda (x_s \dots) e_b), \rho_\lambda) :: v_s \\
u \triangleq tick(\varsigma, 1) & b_i \triangleq alloc(\varsigma, i) \\
\sigma[b \mapsto \mathbf{appappk}(v, e, \rho_\kappa, c)] & u \triangleq tick(\varsigma, n) \\
\\
\mathbf{appappk}(v_f, -, \rho_\kappa, c) \rightsquigarrow E\langle e_b, \rho'_\lambda, c, u \rangle & \rho'_\lambda \triangleq \rho_\lambda[x_0 \mapsto b_0 \dots \\
\text{where } v_f = ((\lambda (x_s \dots) e_b), \rho_\lambda) & \quad x_{n-1} \mapsto b_{n-1}] \\
b_i \triangleq alloc(\varsigma, i) & v'_s \triangleq v_s \# [v] \\
u \triangleq tick(\varsigma, n) & \sigma[b_0 \mapsto v'_0 \dots b_{n-1} \mapsto v'_{n-1}] \\
\rho'_\lambda \triangleq \rho_\lambda[x_0 \mapsto b_0 \dots & \\
\quad x_{n-1} \mapsto b_{n-1}] & \mathbf{appk}(done, [], \rho_\kappa, c) \rightsquigarrow E\langle e_b, \rho'_\lambda, c, u \rangle \\
\sigma[b_0 \mapsto v_0 \dots b_{n-1} \mapsto v_{n-1}] & \text{where } done = ((\lambda x e_b), \rho_\lambda) :: v_s \\
b \triangleq alloc(\varsigma, 0) \\
u \triangleq tick(\varsigma, 1) \\
\rho'_\lambda \triangleq \rho_\lambda[x \mapsto b] \\
v'_s \triangleq (v_s \# [v]) \\
\sigma[b \mapsto v'_s] \\
\\
\mathbf{appappk}(v_f, -, \rho_\kappa, c) \rightsquigarrow E\langle e_b, \rho'_\lambda, c, u \rangle & \mathbf{appk}([\kappa_\lambda], [], \rho_\kappa, c) \rightsquigarrow A\langle v, \rho_\kappa, b, u \rangle \\
\text{where } v_f = ((\lambda x e_b), \rho_\lambda) & \text{where } b \triangleq alloc(\varsigma, 0) \\
b \triangleq alloc(\varsigma, 0) & u \triangleq tick(\varsigma, 1) \\
u \triangleq tick(\varsigma, 1) & \sigma[b \mapsto \kappa_\lambda] \\
\rho'_\lambda \triangleq \rho_\lambda[x \mapsto b] & \\
\sigma[b \mapsto v] & \\
\\
\mathbf{appk}(done, e_h :: e_t, \rho_\kappa, c) & \\
\rightsquigarrow E\langle e_h, \rho_\kappa, b, u \rangle & \\
\text{where } b \triangleq alloc(\varsigma, 0) & \\
u \triangleq tick(\varsigma, 1) & \\
\sigma[b \mapsto \mathbf{appk}(done \# [v], & \\
\quad e_t, \rho_\kappa, c)] &
\end{array}$$

2 Abstract Semantics of Scheme CESK*