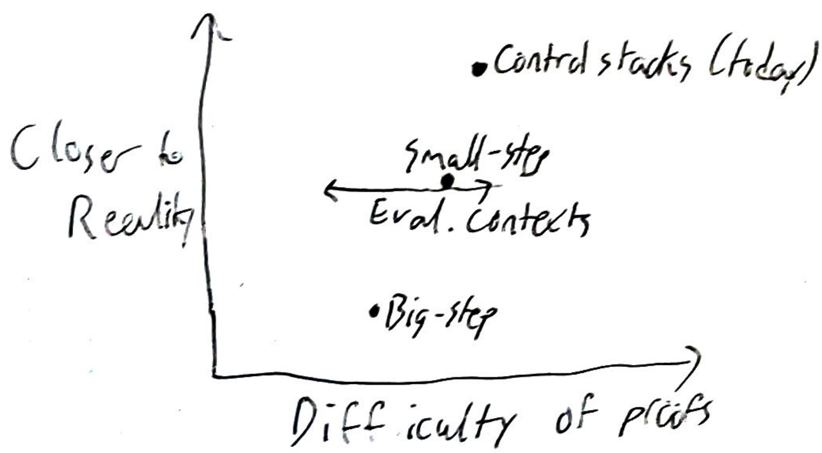
Control Stacks

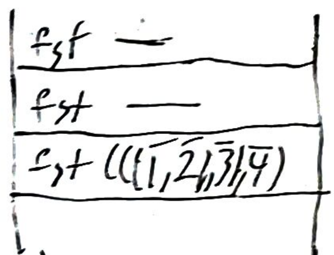


$$\dfrac{\mathsf{fst}\ (\lambda x{:}\mathsf{int}.x, \lambda x{:}\mathsf{int}.x) \mapsto \lambda x{:}\mathsf{int}.x}{(\mathsf{fst}\ (\lambda x{:}\mathsf{int}.x, \lambda x{:}\mathsf{int}.x))\ (\mathsf{snd}\ (\overline{1},\overline{2})) \mapsto (\lambda x{:}\mathsf{int}.x)(\mathsf{snd}\ (\overline{1},\overline{2}))}$$

$$\left.\dfrac{\dfrac{\mathsf{fst}\ (((\overline{1},\overline{2}),\overline{3}),\overline{4}) \mapsto ((\overline{1},\overline{2}),\overline{3})}{\mathsf{fst}\ (\mathsf{fst}\ (((\overline{1},\overline{2}),\overline{3}),\overline{4})) \mapsto \mathsf{fst}\ ((\overline{1},\overline{2}),\overline{3})}}{\mathsf{fst}\ (\mathsf{fst}\ (\mathsf{fst}\ (((\overline{1},\overline{2}),\overline{3}),\overline{4}))) \mapsto \mathsf{fst}\ (\mathsf{fst}\ ((\overline{1},\overline{2}),\overline{3}))}\right\}\ \text{How does a computer remember all this?}$$

The stack:

For us:



Derived from the K Machine (Harper)

Stack frames $\quad f ::= -\ e \mid v\ - \mid (-, e) \mid (v, -) \mid \mathsf{fst}\ - \mid \mathsf{snd}\ -$

Stacks $\quad k ::= \underset{\text{empty}}{\varepsilon} \mid k; f$

Two states: $s ::= k \triangleright e$ $\quad$ Evaluating e

$\qquad\qquad\quad \mid k \triangleleft v$ $\quad$ Finished, returning v to stack e.g. $\mathsf{fst}\ -;\ \mathsf{fst}\ - \rhd \mathsf{fst}\ (((\overline{1},\overline{2}),\overline{3}),\overline{4}) \mapsto \mathsf{fst}\ -;\ \mathsf{fst}\ - \lhd ((\overline{1},\overline{2}),\overline{3})$

$$\frac{v\ \mathsf{val}}{k \rhd v \mapsto k \lhd v}\ (S\text{-}1)$$

$$\frac{}{k \rhd e_1\ e_2 \mapsto k;\ -\ e_2 \rhd e_1}\ (S\text{-}2) \qquad \frac{}{k;\ -\ e_2 \lhd v \mapsto k;\ v\ - \rhd e_2}\ (S\text{-}3)$$

$$\frac{}{k;\ \lambda x{:}\tau.\ e_1\ - \lhd v \mapsto k \rhd [v/x]\,e_1}\ (S\text{-}4)$$

$$\frac{}{k \rhd (e_1, e_2) \mapsto k;\ (-, e_2) \rhd e_1}\ (S\text{-}5) \qquad \frac{}{k;\ (-, e_2) \lhd v_1 \mapsto k;\ (v_1, -) \rhd e_2}\ (S\text{-}6)$$

$$\frac{}{k;\ (v_1, -) \lhd v_2 \mapsto k \lhd (v_1, v_2)}\ (S\text{-}7)$$

$$\frac{}{k \rhd \mathsf{fst}\ e \mapsto k;\ \mathsf{fst}\ - \rhd e}\ (S\text{-}8) \qquad \frac{}{k;\ \mathsf{fst}\ - \lhd (v_1, v_2) \mapsto k \lhd v_1}\ (S\text{-}9)$$

$$\frac{}{k \rhd \mathsf{snd}\ e \mapsto k;\ \mathsf{snd}\ - \rhd e}\ (S\text{-}10) \qquad \frac{}{k;\ \mathsf{snd}\ - \lhd (v_1, v_2) \mapsto k \lhd v_2}\ (S\text{-}11)$$

Type Safety

$f : \tau \rightsquigarrow \tau'$ Frame f expects a τ, returns a τ'

$$\frac{\bullet \vdash e : \tau_1}{-\ e : (\tau_1 \to \tau_2) \rightsquigarrow \tau_2}\ (F\text{-}1) \qquad \frac{\bullet \vdash v : \tau_1 \to \tau_2}{v\ - : \tau_1 \rightsquigarrow \tau_2}\ (F\text{-}2) \qquad \frac{\bullet \vdash e_2 : \tau_2}{(-, e_2) : \tau_1 \rightsquigarrow \tau_1 \times \tau_2}\ (F\text{-}3)$$

$$\frac{\bullet \vdash v_1 : \tau_1}{(v_1, -) : \tau_2 \rightsquigarrow \tau_1 \times \tau_2}\ (F\text{-}4) \qquad \frac{}{\mathsf{fst}\ - : \tau_1 \times \tau_2 \rightsquigarrow \tau_1}\ (F\text{-}5) \qquad \frac{}{\mathsf{snd}\ - : \tau_1 \times \tau_2 \rightsquigarrow \tau_2}\ (F\text{-}6)$$

$k \lhd \tau$ k expects a τ

$$\frac{}{\varepsilon \lhd \tau}\ (K\text{-}1) \qquad \frac{k \lhd \tau' \quad f : \tau \rightsquigarrow \tau'}{k;\ f \lhd \tau}\ (K\text{-}2)$$

s ok.

$$\frac{k \lhd \tau \quad \cdot \vdash e : \tau}{k \rhd e \ \text{ok}} \text{(OK-1)} \qquad \frac{k \lhd \tau \quad \cdot \vdash v : \tau \quad (v\ \text{val})}{k \lhd v \ \text{ok}} \text{(OK-2)}$$

Preservation: If s ok and $s \mapsto s'$ then s' ok.

Proof: By "induction" on the derivation of $s \mapsto s'$

S-1 By inversion on (OK-1), $k \lhd \tau$ and $\cdot \vdash v : \tau$. Apply (OK-2)

S-2. Then $s = k \rhd e_1\ e_2$ and $s' = k; -\ e_2 \rhd e_1$.

By inversion, $k \lhd \tau$ and $\cdot \vdash e_1\ e_2 : \tau$. By inversion on $(\to E)$, $\cdot \vdash e_1 : \tau_1 \to \tau$ and $\cdot \vdash e_2 : \tau_1$. By (F-1) and (K-2), $k; -\ e_2 : (\tau_1 \to \tau) \rightsquigarrow \tau$.

Apply (OK-1).

S-3. Then $s = k; -\ e_2 \lhd v$ and $s' = k; v - \rhd e_2$.

By inversion, $k \lhd \tau_2$ and $e_2 : \tau_1$ and $\cdot \vdash v . \tau_1 \rightsquigarrow \tau_2$

By (F-2), $v - : \tau_1 \rightsquigarrow \tau_2$. By K-2, $k; v - \lhd \tau_1$. Apply (OK-2).

S-4. Then $s = k; \lambda x{:}\tau.e_1 - \lhd v$ and $s' = k \rhd [v/x]e_1$.

By inversion, $k \lhd \tau_2$ and $\cdot \vdash v : \tau_1$ and $\cdot \vdash \lambda x{:}\tau.e_1 : \tau_1 \to \tau_2$.

By inversion on $(\to I)$, $x : \tau_1 \vdash e_1 : \tau_2$.

By substitution, $\cdot \vdash [v/x]\, e_1 : \tau_2$. Apply (OK-1) □

Progress: If s ok, then $s = \varepsilon \lhd v$ or $\exists s'$ such that $s \mapsto s'$.

Proof: By "induction" on the derivation of s ok.

(OK-1) Then $s = k \rhd e$ and $k \lhd \tau$ and $\cdot \vdash e : \tau$.

Proceed by nested "induction" on the derivation of $\cdot \vdash e : \tau$.

(unit-I), $(\to I)$ Then e val. Apply (S-1).

$(\to E)$. Then $e = e_1\ e_2$. (S-2)

$(\times I)$. (S-9)

$(\times E_1)$. (S-8)

$(\times E_2)$. (S-10).

(OK-2). Then $s = k \lhd v$ and $k \lhd \tau$ and $\cdot \vdash e : \tau$.

(K-1). Then $s = \varepsilon \lhd v$.

(K-2). Then $k = k_0; f$ and $k_0 \lhd \tau'$ and $f : \tau \rightsquigarrow \tau'$.

(F-1). Then $f = - \; e$ and $\bullet \vdash e : \tau_1$ and $\tau = \tau_1 \to \tau'$. (S-3).
(F-2). Then $f = v_1 \; -$ and $\bullet \vdash v : \tau \to \tau'$.
By CF, $v_1 = \lambda x : \tau . \; e_1$. (S-4)
(F-3). Then $f = (-, e_2)$ and $\bullet \vdash e : \tau_2$ and $\tau' = \tau \times \tau_2$. (S-6)
(F-4) Then $f = (v, -)$. (S-7)
(F-5). Then $f = fst \; -$ and $\tau = \tau_1 \times \tau_2$. By CF, $v = (v_1, v_2)$. (S-9) $\square$