SCC machine $\langle M, E \rangle$

$$\langle (M\ N), E \rangle \longmapsto_{scc} \langle M, E[(\square\ N)] \rangle$$

$$\langle V, E[(\square\ N)] \rangle \longmapsto_{scc} \langle N, E[(V\ \square)] \rangle$$

$$\langle V, E[(\lambda x. M)\ \square] \rangle \longmapsto_{scc} \langle M[X \leftarrow V], E \rangle$$

Normal → Standard → CC → SCC

$2^n$ $\quad 4n^2 \quad 2n^2 \quad 1n^2$

$E = \square$ — new Hole();

$\quad |\ (E\ M)$ — new OnLeft( EvCt E, Term M)

$\quad |\ (V\ E)$ — new OnRight ( ~~Val~~ V, EvCt E)

$\Downarrow$

$E' = $ listof (Frame)

Frame $= (\square\ M)$ — new LeftFrame (Term M)

$\quad |\ (V\ \square)$ — new RightFrame (Val V)

$E_0 = (((\square\ M)\ N)\ P)$

$E_0'^R = (\square\ M)\ ;\ (\square\ N)\ ;\ (\square\ P)$

$E_0' = (\square\ P)\ ;\ (\square\ N)\ ;\ (\square\ M)$

en : $E \to E'$ $\qquad$ de : $E' \to E$

$en(\square) = \perp$

$en(E\ M) = \cancel{en(E)}\ (\square\ M)\ ; \cancel{E}\ en(E)$

$en(V\ E) = (V\ \square)\ ;\ en(E)$

$de(\perp) = \square$

$de((\square\ M)\ ;\ E') = (\square\ M)[de(E')] = (de(E')\ M)$

$de((V\ \square);\ E') = (V\ de(E'))$

8-2

$SCC'$ $\quad \langle M, E' \rangle$

$O(n)$ $\quad \langle (M\ N), E' \rangle \longmapsto_{scc'} \langle M, E' \mathbin{++} (\boxtimes\ N) \rangle$

$2 \cdot O(n)$ $\quad \langle V, E' \mathbin{++} (\boxtimes\ N) \rangle \longmapsto_{scc'} \langle N, E' \mathbin{++} (V\ \boxtimes) \rangle$

$2 \cdot O(n)$ $\quad \langle V, E' \mathbin{++} ((\lambda X.M)\ \boxtimes) \rangle \longmapsto_{scc'} \langle M[X \leftarrow V], E' \rangle$

$$\begin{tikzcd}
\langle M_0, E_0 \rangle \arrow[r, "scc" ', |->>] \arrow[d, "en"] & \langle M_1, E_1 \rangle \\
\langle M_0, E'_0 \rangle \arrow[r, "scc'" ', |->>] & \langle M_1, E'_1 \rangle \arrow[u, "de" ']
\end{tikzcd}$$

$SCC'^R$ $\quad \langle M, E'^R \rangle$

$O(1)$ $\quad \langle (M\ N), E'^R \rangle \longmapsto_{scc'R} \langle M, (\boxtimes\ N); E'^R \rangle$

$2 \cdot O(1)$ $\quad \langle V, (\boxtimes\ N); E'^R \rangle \longmapsto \langle N, (V\ \boxtimes); E'^R \rangle$

$O(1) + O(n)$ $\quad \langle V, ((\lambda X.M)\ \boxtimes); E'^R \rangle \longmapsto \langle M[X \leftarrow V], E'^R \rangle$

$$\begin{tikzcd}
\langle M_0, E'_0 \rangle \arrow[r, |->>] \arrow[d, "rev"] & \langle M_1, E_1 \rangle \\
\langle M_0, E'^R_0 \rangle \arrow[r, |->>] & \langle M_1, E'^R_1 \rangle \arrow[u, "rev" ']
\end{tikzcd}$$

$en^R : E \to E'^R$ $\qquad en^R = en \circ rev$

$en^R(\boxtimes) = \perp$

$en^R((E\ N)) = (\cancel{\boxtimes\ N); } en^R(E) \mathbin{++} (\boxtimes\ N)$

$en^R((V\ E)) = (\cancel{V\ \boxtimes); } en^R(E) \mathbin{++} (V\ \boxtimes)$

$de^R : E'^R \Rightarrow E$

$de^R(\perp) = \boxtimes$

$de^R((\boxtimes\ N); E'^R) = de(E'^R)[(\boxtimes\ N)]$

$de^R((V\ \boxtimes); E'^R = de(E'^R)[(V\ \boxtimes)]$

8-4

$M = \ldots$
$\quad \mid \; (\text{callcc } M)$

$\langle (\text{callcc } M), \kappa \rangle$
$\quad \longmapsto \langle M, \text{fun}(\underline{\kappa}, \kappa) \rangle$

$\langle \kappa, \text{arg}(\lambda X. M, \kappa') \rangle \longmapsto \langle M[X \leftarrow \kappa], \kappa' \rangle$

$\langle V, \text{arg}(\kappa, \kappa') \rangle \longmapsto \langle V, \kappa \rangle$

$(+\ 1\ (\text{callcc}\ (\lambda k.\ (+\ 2\ (k\ 3)))))$

$(+\ 2)$ $\quad (+\ 1\ (\text{ret}))$

$(+\ 1\ 3)$

$\Rightarrow 4$

---

every language has continuation
some langs can implement them with stacks

the langs that can't, typically have the feature "first-class continuations"

---

$\kappa = \ldots \mid \text{prim}(o^n, V\ldots, M\ldots, \kappa)$

$\langle (o^n\, N\, M\ldots), \kappa \rangle$
$\quad \longmapsto \langle N, \text{prim}(o^n, \bot, M\ldots, \kappa) \rangle$

$\langle V, \text{prim}(o^n, V'\ldots, N\, M\ldots, \kappa) \rangle$
$\quad \longmapsto \langle N, \text{prim}(o^n, V; V'\ldots, M\ldots, \kappa) \rangle$

$\langle V, \text{prim}(o^n, V'\ldots, \bot, \kappa) \rangle$
$\quad \longmapsto \langle \delta(o^n, (V; V'\ldots)^R), \kappa \rangle$

8-5

CK - machine $\langle M, K \rangle$

C = control string

K = ... Stac**K** *

"continuation"

**k**ontinuation

$K = \text{ret}$

$\quad \mid \text{fun}(N, K)$

$\quad \mid \text{arg}(V, K)$

| | L | | R |
|---|---|---|---|
| 1. | $\langle (M\ N), K \rangle$ | $\longmapsto_{ck}$ | $\langle M, \text{fun}(N, K) \rangle$ |
| 2. | $\langle V, \text{fun}(N, K) \rangle$ | $\longmapsto_{ck}$ | $\langle N, \text{arg}(V, K) \rangle$ |
| 3. | $\langle V, \text{arg}(\lambda X.M, K) \rangle$ | $\longmapsto_{ck}$ | $\langle M[X \leftarrow V], K \rangle$ |

ASM

$\vdots$

$\underbrace{\text{arr[17]}}$ ( $\underbrace{x+21}$ ) → register 3

requires 4 register

arr * x + y * z + q

C

$\vdots$

; compute f
push r3
use r0 ... r3
remember f in r2
pop r3
; compute arg
push r2
....
arg is in r1
pop r2

ret = $\lambda r.\lambda f.\lambda a.\lambda x.\ r x$

fun $\quad f x$

arg $\quad a x$