Natural semantics

$$(x := a, s) \to s[x \mapsto \mathcal{A}[a]s]$$

$$(\texttt{skip}, s) \to s$$

$$\frac{(S_1, s) \to s', \quad (S_2, s') \to s''}{(S_1; S_2, s) \to s''}$$

$$\frac{(S_1, s) \to s'}{(\texttt{if } b \texttt{ then } S_1 \texttt{ else } S_2, s) \to s'}$$

$$\text{if } \mathcal{B}[b]s = \mathsf{tt}$$

$$\frac{(S_2, s) \to s'}{(\texttt{if } b \texttt{ then } S_1 \texttt{ else } S_2, s) \to s'}$$

$$\text{if } \mathcal{B}[b]s = \mathsf{ff}$$

$$\frac{(S, s) \to s', \quad (\texttt{while } b \texttt{ do } S, s') \to s''}{(\texttt{while } b \texttt{ do } S, s) \to s''}$$

$$\text{if } \mathcal{B}[b]s = \mathsf{tt}$$

$$(\texttt{while } b \texttt{ do } S, s) \to s$$

$$\text{if } \mathcal{B}[b]s = \mathsf{ff}$$

Blocks and variable declarations

Extension of semantics of statements:

$$\frac{(D_V,\, s) \rightarrow_D s',\ (S,\, s') \rightarrow s''}{(\texttt{begin } D_V\ S \texttt{ end},\, s) \rightarrow s''[\mathrm{DV}(D_V) \longmapsto s]}$$

Semantics of variable declarations:

Typical transition:

$$(D_V,\, s) \rightarrow_D s'$$

defined by

$$\frac{(D_V,\, s[x \mapsto \mathcal{A}[a]s]) \rightarrow_D s'}{(\texttt{var } x := a;\, D_V,\, s) \rightarrow_D s'}$$

$$(\varepsilon,\, s) \rightarrow_D s$$

Scope rules

- dynamic scope for variables as well as procedures
- dynamic scope for variables but static scope for procedures
- static scope for variables as well as procedures

```
begin var x := 0;
      proc p is x := x * 2;
      proc q is call p;
      begin var x := 5;
            proc p is x := x + 1;
            call q; y := x
      end
end
```

Dynamic scope rules:

$$env_P \in \mathsf{Env}_P = \mathsf{Pname} \hookrightarrow \mathsf{Stm}$$

Static scope rules for variables:

$$env_V \in \mathsf{Env}_V = \mathsf{Var} \rightarrow \mathsf{Loc}$$

$$sto \in \mathsf{Store} = \mathsf{Loc} \cup \{ \mathsf{next} \} \rightarrow \mathsf{Z}$$

Static scope rules for procedures:

$$env_P \in \mathsf{Env}_P = \mathsf{Pname} \hookrightarrow \mathsf{Stm} \times \mathsf{Env}_V \times \mathsf{Env}_P$$

$$\boxed{env_P \vdash (S,\ s) \to s'}$$

$$env_P \vdash (x := a,\ s) \to s[x \mapsto \mathcal{A}[a]s]$$

$$env_P \vdash (\texttt{skip},\ s) \to s$$

$$\frac{env_P \vdash (S_1,\ s) \to s',\ env_P \vdash (S_2,\ s') \to s''}{env_P \vdash (S_1;S_2,\ s) \to s''}$$

$$\frac{env_P \vdash (S_1,\ s) \to s'}{env_P \vdash (\texttt{if}\ b\ \texttt{then}\ S_1\ \texttt{else}\ S_2,\ s) \to s'}$$

if $\mathcal{B}[b]s = \mathsf{tt}$

$$\frac{env_P \vdash (S_2,\ s) \to s'}{env_P \vdash (\texttt{if}\ b\ \texttt{then}\ S_1\ \texttt{else}\ S_2,\ s) \to s'}$$

if $\mathcal{B}[b]s = \mathsf{ff}$

$$\frac{\begin{array}{c} env_P \vdash (S,\ s) \to s', \\ env_P \vdash (\texttt{while}\ b\ \texttt{do}\ S,\ s') \to s'' \end{array}}{env_P \vdash (\texttt{while}\ b\ \texttt{do}\ S,\ s) \to s''}$$

if $\mathcal{B}[b]s = \mathsf{tt}$

$$env_P \vdash (\texttt{while}\ b\ \texttt{do}\ S,\ s) \to s$$

if $\mathcal{B}[b]s = \mathsf{ff}$

$env_P \vdash (S,\, s) \to s'$ (cont.)

$$\frac{\begin{array}{c}(D_V,\, s) \to_D s', \\ \mathrm{upd}_P(D_P,\, env_P) \vdash (S,\, s') \to s''\end{array}}{\begin{array}{c}env_P \vdash (\texttt{begin}\ D_V\ D_P\ S\ \texttt{end},\, s) \to \\ s''[\mathrm{DV}(D_V) \longmapsto s]\end{array}}$$

$$\frac{env_P \vdash (S,\, s) \to s'}{env_P \vdash (\texttt{call}\ p,\, s) \to s'}$$

where $env_P\ p = S$

$\mathsf{upd}_P(\texttt{proc}\ p\ \texttt{is}\ S;\ D_P,\, env_P) =$
$\qquad \mathsf{upd}_P(D_P,\, env_P[p \mapsto S])$

$\mathsf{upd}_P(\varepsilon,\, env_P) = env_P$

$$\boxed{(D_V,\, env_V,\, sto) \rightarrow_D (env'_V,\, sto')}$$

$$\frac{\begin{array}{c}(D_V,\, env_V[x \mapsto l],\, sto[l \mapsto v][\mathrm{next} \mapsto l'])\\ \rightarrow_D (env'_V,\, sto')\end{array}}{(\mathtt{var}\ x := a;\ D_V,\, env_V,\, sto) \rightarrow_D (env'_V,\, sto')}$$

where $v = \mathcal{A}[a](sto \circ env_V)$
and $l = sto$ next
and $l' =$ new l

$$(\varepsilon,\, env_V,\, sto) \rightarrow_D (env_V,\, sto)$$

$$\boxed{env_V,\ env_P \vdash (S,\ sto) \to sto'}$$

$$env_V,\ env_P \vdash (x := a,\ sto) \to sto[l \mapsto v]$$

where $l = env_V\ x$
and $v = \mathcal{A}[a](sto \circ env_V)$

$$env_V,\ env_P \vdash (\texttt{skip},\ sto) \to sto$$

$$\frac{\begin{array}{c} env_V,\ env_P \vdash (S_1,\ sto) \to sto', \\ env_V,\ env_P \vdash (S_2,\ sto') \to sto'' \end{array}}{env_V,\ env_P \vdash (S_1;S_2,\ sto) \to sto''}$$

$$\frac{env_V,\ env_P \vdash (S_1,\ sto) \to sto'}{\begin{array}{c} env_V,\ env_P \vdash (\texttt{if}\ b\ \texttt{then}\ S_1\ \texttt{else}\ S_2,\ sto) \\ \to sto' \end{array}}$$

if $\mathcal{B}[b](sto \circ env_V) = \mathsf{tt}$

$$\frac{env_V,\ env_P \vdash (S_2,\ sto) \to sto'}{\begin{array}{c} env_V,\ env_P \vdash (\texttt{if}\ b\ \texttt{then}\ S_1\ \texttt{else}\ S_2,\ sto) \\ \to sto' \end{array}}$$

if $\mathcal{B}[b](sto \circ env_V) = \mathsf{ff}$

$$\boxed{env_V,\ env_P \vdash (S,\ sto) \to sto' \text{ (cont.)}}$$

$$\frac{\begin{array}{c} env_V,\ env_P \vdash (S,\ sto) \to sto', \\ env_V,\ env_P \vdash (\texttt{while}\ b\ \texttt{do}\ S,\ sto') \to sto'' \end{array}}{env_V,\ env_P \vdash (\texttt{while}\ b\ \texttt{do}\ S,\ sto) \to sto''}$$

if $\mathcal{B}[b](sto \circ env_V) = \mathsf{tt}$

$$env_V,\ env_P \vdash (\texttt{while}\ b\ \texttt{do}\ S,\ sto) \to sto$$

if $\mathcal{B}[b](sto \circ env_V) = \mathsf{ff}$

$$\frac{\begin{array}{c} (D_V,\ env_V,\ sto) \to_D (env'_V,\ sto'), \\ env'_V,\ env'_P \vdash (S,\ sto') \to sto'' \end{array}}{\begin{array}{c} env_V,\ env_P \vdash (\texttt{begin}\ D_V\ \ D_P\ S\ \texttt{end},\ sto) \\ \to sto'' \end{array}}$$

where $env'_P = \mathsf{upd}_P(D_P,\ env'_V,\ env_P)$

$$\begin{array}{l} \mathsf{upd}_P(\texttt{proc}\ p\ \texttt{is}\ S;\ D_P,\ env_V,\ env_P) = \\ \qquad \mathsf{upd}_P(D_P,\ env_V, \\ \qquad\qquad env_P[p \mapsto (S, env_V, env_P)]) \end{array}$$

$$\mathsf{upd}_P(\varepsilon,\ env_V,\ env_P) = env_P$$

env_V, $env_P \vdash (S,\ sto) \rightarrow sto'$ (cont.)

$$\frac{env'_V,\ env'_P \vdash (S,\ sto) \rightarrow sto'}{env_V,\ env_P \vdash (\texttt{call}\ p,\ sto) \rightarrow sto'}$$

where $env_P\ p = (S,\ env'_V,\ env'_P)$

$$\frac{\begin{array}{c} env'_V,\ env'_P[p \mapsto (S,\ env'_V,\ env'_P)] \\ \vdash (S,\ sto) \rightarrow sto' \end{array}}{env_V,\ env_P \vdash (\texttt{call}\ p,\ sto) \rightarrow sto'}$$

where $env_P\ p = (S,\ env'_V,\ env'_P)$