
Small Step Semantics

Transition format

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle$$

$$P \vdash \langle es, s \rangle [\rightarrow] \langle es', s' \rangle$$

Reduction strategy:

Reduce subexpressions as long as necessary,
then reduce whole expression.

Two kinds of rules

- Subexpression reduction (for construct c):

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies \\ P \vdash \langle c \dots e \dots, s \rangle \rightarrow \langle c \dots e' \dots, s' \rangle$$

Example:

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies \\ P \vdash \langle \text{Cast } C e, s \rangle \rightarrow \langle \text{Cast } C e', s' \rangle$$

- Expression reduction

Example: $P \vdash \langle \text{Cast } C \text{ null}, s \rangle \rightarrow \langle \text{null}, s \rangle$

Notation: state components

hp \equiv *fst*

lcl \equiv *snd*

Val and Var

$|cl\ s\ V = [v] \implies$

$P \vdash \langle \text{Var } V, s \rangle \rightarrow \langle \text{Val } v, s \rangle$

Field access

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies$$

$$P \vdash \langle e.F\{D\}, s \rangle \rightarrow \langle e'.F\{D\}, s' \rangle$$

$$\llbracket hp \ s \ a = \lfloor (C, fs) \rfloor; fs \ (F, D) = \lfloor v \rfloor \rrbracket$$

$$\implies P \vdash \langle addr \ a.F\{D\}, s \rangle \rightarrow \langle Val \ v, s \rangle$$

Binary operations

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies$$

$$P \vdash \langle e \ll bop \gg e_2, s \rangle \rightarrow \langle e' \ll bop \gg e_2, s' \rangle$$

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies$$

$$P \vdash \langle \text{Val } v_1 \ll bop \gg e, s \rangle \rightarrow \langle \text{Val } v_1 \ll bop \gg e', s' \rangle$$

$$\text{binop } (bop, v_1, v_2) = [v] \implies$$

$$P \vdash \langle \text{Val } v_1 \ll bop \gg \text{Val } v_2, s \rangle \rightarrow \langle \text{Val } v, s \rangle$$

new

$\llbracket \text{new-Addr } h = \lfloor a \rfloor ;$

$P \vdash C \text{ has-fields FDTs};$

$h' = h(a \mapsto (C, \text{init-fields FDTs})) \rrbracket$

$\implies P \vdash \langle \text{new } C, (h, I) \rangle \rightarrow \langle \text{addr } a, (h', I) \rangle$

Cast

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies$$
$$P \vdash \langle \text{Cast } C \ e, s \rangle \rightarrow \langle \text{Cast } C \ e', s' \rangle$$
$$\llbracket hp \ s \ a = \lfloor (D, fs) \rfloor; P \vdash D \preceq^* C \rrbracket$$
$$\implies P \vdash \langle \text{Cast } C \ (\text{addr } a), s \rangle \rightarrow \langle \text{addr } a, s \rangle$$
$$P \vdash \langle \text{Cast } C \ \text{null}, s \rangle \rightarrow \langle \text{null}, s \rangle$$

Variable assignment

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies$$

$$P \vdash \langle V := e, s \rangle \rightarrow \langle V := e', s' \rangle$$

$$P \vdash \langle V := \text{val } v, (h, l) \rangle \rightarrow \langle \text{unit}, (h, l(V \mapsto v)) \rangle$$

Field assignment

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies$$

$$P \vdash \langle e.F\{D\} := e_2, s \rangle \rightarrow \langle e'.F\{D\} := e_2, s' \rangle$$

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies$$

$$P \vdash \langle \text{Val } v.F\{D\} := e, s \rangle \rightarrow \langle \text{Val } v.F\{D\} := e', s' \rangle$$

$$h a = \lfloor (C, fs) \rfloor \implies$$

$$P \vdash \langle \text{addr } a.F\{D\} := \text{Val } v, (h, I) \rangle \rightarrow \\ \langle \text{unit}, (h(a \mapsto (C, fs((F, D) \mapsto v))), I) \rangle$$

Method call

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies P \vdash \langle e.M(es), s \rangle \rightarrow \langle e'.M(es), s' \rangle$$

$$P \vdash \langle es, s \rangle [\rightarrow] \langle es', s' \rangle \implies$$

$$P \vdash \langle \text{Val } v.M(es), s \rangle \rightarrow \langle \text{Val } v.M(es'), s' \rangle$$

$$\llbracket hp \ s \ a = \lfloor (C, fs) \rfloor; P \vdash C \text{ sees } M: Ts \rightarrow T = (pns, body) \text{ in } D;$$

$$\llbracket |vs| = |pns|; |Ts| = |pns| \rrbracket$$

$$\implies P \vdash \langle \text{addr } a.M(\text{map } \text{Val } vs), s \rangle \rightarrow$$

$$\langle \text{blocks}(\text{this} \cdot pns, \text{Class } D \cdot Ts, \text{Addr } a \cdot vs, body), s \rangle$$

$$\text{blocks}([V, \dots], [T, \dots], [v, \dots], e) = \{ V:T; V := \text{Val } v; \dots e \}$$

$$P \vdash \langle es, s \rangle [\rightarrow] \langle es', s' \rangle$$

Evaluate es from left to right:

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies$$

$$P \vdash \langle e \cdot es, s \rangle [\rightarrow] \langle e' \cdot es, s' \rangle$$

$$P \vdash \langle es, s \rangle [\rightarrow] \langle es', s' \rangle \implies$$

$$P \vdash \langle \text{Val } v \cdot es, s \rangle [\rightarrow] \langle \text{Val } v \cdot es', s' \rangle$$

Local variable

Trick: store value of local variable as $\{V:T; V := \text{val } v; e'\}$

assigned $V e \equiv \exists v e'. e = V := \text{val } v; e'$

$\llbracket P \vdash \langle e, (h, I(V := \text{None})) \rangle \rrbracket \rightarrow \langle e', (h', I') \rangle; I' V = \text{None};$

$\neg \llbracket \text{assigned } V e \rrbracket$

$\implies P \vdash \langle \{V:T; e\}, (h, I) \rangle \rightarrow \langle \{V:T; e'\}, (h', I'(V := I V)) \rangle$

$\llbracket P \vdash \langle e, (h, I(V := \text{None})) \rangle \rrbracket \rightarrow \langle e', (h', I') \rangle; I' V = [v];$

$\neg \llbracket \text{assigned } V e \rrbracket$

$\implies P \vdash \langle \{V:T; e\}, (h, I) \rangle \rightarrow$

$\langle \{V:T; V := \text{val } v; e'\}, (h', I'(V := I V)) \rangle$

Local variable (2)

$$\begin{aligned} & \llbracket P \vdash \langle e, (h, l(V \mapsto v)) \rangle \rightarrow \langle e', (h', l') \rangle; l' V = [v'] \rrbracket \\ \implies & P \vdash \langle \{V:T; V := \text{val } v; e\}, (h, l) \rangle \rightarrow \\ & \langle \{V:T; V := \text{val } v'; e'\}, (h', l'(V := l V)) \rangle \end{aligned}$$

$$P \vdash \langle \{V:T; V := \text{val } v; \text{val } u\}, s \rangle \rightarrow \langle \text{val } u, s \rangle$$

$$P \vdash \langle \{V:T; \text{val } u\}, s \rangle \rightarrow \langle \text{val } u, s \rangle$$

Sequential composition

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies$$

$$P \vdash \langle e; e_2, s \rangle \rightarrow \langle e'; e_2, s' \rangle$$

$$P \vdash \langle \text{Val } v; e_2, s \rangle \rightarrow \langle e_2, s \rangle$$

Conditional

$$P \vdash \langle e, s \rangle \rightarrow \langle e', s' \rangle \implies$$
$$P \vdash \langle \text{if } (e) \ e_1 \ \text{else } e_2, s \rangle \rightarrow \langle \text{if } (e') \ e_1 \ \text{else } e_2, s' \rangle$$
$$P \vdash \langle \text{if } (\text{true}) \ e_1 \ \text{else } e_2, s \rangle \rightarrow \langle e_1, s \rangle$$
$$P \vdash \langle \text{if } (\text{false}) \ e_1 \ \text{else } e_2, s \rangle \rightarrow \langle e_2, s \rangle$$

While loop

$P \vdash \langle \text{while } (b) \ c, s \rangle \rightarrow$
 $\langle \text{if } (b) \ (c; \text{while } (b) \ c) \ \text{else } \text{unit}, s \rangle$

Example

$$P \vdash \langle \{ V: \text{Boolean}; U := \text{true}; V := \text{Var } U; \text{Var } V \}, \\ (h, \text{empty}) \rangle \rightarrow^* \\ \langle \text{true}, (h, [U \mapsto \text{Bool True}]) \rangle$$

Example

```
class C extends Object
{ field F:Integer
  method m(i:Integer):Integer =
    this.F{C} + Var i
}
```

State:

$$h = [a \mapsto (C, fs)]$$
$$fs = [(F, C) \mapsto \text{Intg } 2]$$
$$l = [V \mapsto \text{Addr } a]$$

Reduction:

$$P \vdash \langle \text{Var } V.m([\text{val } (\text{Intg } 1)]), (h, l) \rangle \rightarrow^* \langle \text{val } (\text{Intg } 3), (h, l) \rangle$$