

# 1 Syntax

$program ::= \overline{cls} \ \overline{s}$   
 $cls ::= \text{class } C \ \{\overline{field} \ \overline{method}\}$   
 $field ::= T \ f;$   
 $method ::= T \ m(\overline{T} \ x) \ \text{contract} \ \{\overline{s}\}$   
 $contract ::= \text{requires } \phi; \ \text{ensures } \phi;$   
 $T ::= \text{int} \mid C$   
 $s ::= x.f := y; \mid x := e; \mid x := \text{new} C; \mid x := y.m(\overline{z}); \mid \text{return } x; \mid \text{assert } \phi; \mid \text{release } \phi;$   
 $\phi ::= \text{true} \mid e = e \mid e \neq e \mid \text{acc}(x.f) \mid \phi * \phi$   
 $e ::= v \mid x \mid e.f$   
 $x ::= \text{this} \mid \text{result} \mid \langle other \rangle$

$\Gamma ::= (x \mapsto T)$   
 $H ::= (o \mapsto (C, (\overline{f \mapsto v})))$   
 $\rho ::= (x \mapsto v)$   
 $A_s ::= \overline{(x, f)}$   
 $A_d ::= \overline{(o, f)}$   
 $S ::= (\rho, A_d, \overline{s}) \cdot S \mid nil$

## 2 Static semantics

### 2.1 Expressions ( $A_s \vdash_{\text{sfrm}} e$ )

$$\frac{}{A \vdash_{\text{sfrm}} x} \text{WFVAR}$$

$$\frac{}{A \vdash_{\text{sfrm}} v} \text{WFVALUE}$$

$$\frac{(x, f) \in A}{A \vdash_{\text{sfrm}} x.f} \text{WFFIELD}$$

### 2.2 Formulas ( $A_s \vdash_{\text{sfrm}} \phi$ )

$$\frac{}{A \vdash_{\text{sfrm}} \text{true}} \text{WFTTRUE}$$

$$\frac{A \vdash_{\text{sfrm}} e_1 \quad A \vdash_{\text{sfrm}} e_2}{A \vdash_{\text{sfrm}} e_1 = e_2} \text{WFEQUAL}$$

$$\frac{A \vdash_{\text{sfrm}} e_1 \quad A \vdash_{\text{sfrm}} e_2}{A \vdash_{\text{sfrm}} e_1 \neq e_2} \text{WFNEQUAL}$$

$$\frac{}{A \vdash_{\text{sfrm}} \text{acc}(x, f)} \text{WFAcc}$$

$$\frac{A_s \vdash_{\text{sfrm}} \phi_1 \quad A_s \cup \text{static-footprint}(\phi_1) \vdash_{\text{sfrm}} \phi_2}{A_s \vdash_{\text{sfrm}} \phi_1 * \phi_2} \text{WF-SepOp}$$

### 2.3 Footprint ( $\text{static-footprint}(\phi) = A_s$ )

$$\begin{aligned} \text{static-footprint}(\text{true}) &= \emptyset \\ \text{static-footprint}(e_1 = e_2) &= \emptyset \\ \text{static-footprint}(e_1 \neq e_2) &= \emptyset \\ \text{static-footprint}(\text{acc}(x.f)) &= \{(x, f)\} \\ \text{static-footprint}(\phi_1 * \phi_2) &= \text{static-footprint}(\phi_1) \cup \text{static-footprint}(\phi_2) \end{aligned}$$

### 2.4 Hoare ( $\Gamma \vdash \{\phi\} \bar{s} \{\phi\}$ )

$$\frac{\Gamma \vdash \{\phi_p\} s_1 \{\phi_{q1}\} \quad \phi_{q1} \implies \phi_{q2} \quad \Gamma \vdash \{\phi_{q2}\} s_2 \{\phi_r\}}{\Gamma \vdash \{\phi_p\} s_1; s_2 \{\phi_r\}} \text{H-Sec}$$

$$\frac{\Gamma(x) = C \quad \text{fields}(C) = \{\bar{f}_i\}}{\Gamma \vdash \{\phi\} x := \text{new } C \{\text{acc}(x.f_i) * x \neq \text{null} * \phi\}} \text{H-NewObj}$$

$$\frac{\phi \implies \text{acc}(x.f) * x \neq \text{null}}{\Gamma \vdash \{\phi\} x.f := y \{\phi * x.f = y\}} \text{H-FieldAssign}$$

$$\frac{\phi' = \phi[e/x] \quad \emptyset \vdash_{\text{sfrm}} \phi' \quad \text{static-footprint}(\phi') \vdash_{\text{sfrm}} e}{\Gamma \vdash \{\phi'\} x := e \{\phi\}} \text{H-VarAssign}$$

$$\begin{array}{c}
\frac{}{\Gamma \vdash \{\phi\} \mathbf{return} \ x \{ \phi * \mathbf{result} = x \}} \quad \text{H-Return} \\
\\
\frac{\Gamma(y) = C \quad \phi \implies y \neq \mathbf{null} * \phi_p * \phi_r \quad \phi_p = \mathbf{mpre}(C, m)[y, \bar{z}/\mathbf{this}, \bar{X}] \quad \phi_q = \mathbf{mpost}(C, m)[y, \bar{z}, x/\mathbf{this}, \bar{X}]}{\Gamma \vdash \{\phi\} x := y.m(\bar{z}) \{ \phi_q * \phi_r \}} \\
\\
\frac{\phi \implies \phi'}{\Gamma \vdash \{\phi\} \mathbf{assert} \ \phi' \{ \phi \}} \quad \text{H-Assert} \\
\\
\frac{\phi \implies \phi' * \phi_r \quad \emptyset \vdash_{\mathbf{sfrm}} \phi_r}{\Gamma \vdash \{\phi\} \mathbf{release} \ \phi' \{ \phi_r \}} \quad \text{H-Release}
\end{array}$$

### 3 Dynamic semantics

#### 3.1 Expressions ( $H, \rho \vdash e \Downarrow v$ )

$$\begin{array}{c}
\frac{}{H, \rho \vdash x \Downarrow \rho(x)} \quad \text{EE-Var} \\
\\
\frac{}{H, \rho \vdash v \Downarrow v} \quad \text{EE-Value} \\
\\
\frac{H, \rho \vdash x \Downarrow o}{H, \rho \vdash x.f \Downarrow H(o)(f)} \quad \text{EE-Acc}
\end{array}$$

#### 3.2 Formulas ( $H, \rho, A \models \phi$ )

$$\begin{array}{c}
\frac{}{H, \rho, A \models \mathbf{true}} \quad \text{EA-True} \\
\\
\frac{H, \rho \vdash e_1 \Downarrow v_1 \quad H, \rho \vdash e_2 \Downarrow v_2 \quad v_1 = v_2}{H, \rho, A \models e_1 = e_2} \quad \text{EA-Equal} \\
\\
\frac{H, \rho \vdash e_1 \Downarrow v_1 \quad H, \rho \vdash e_2 \Downarrow v_2 \quad v_1 = v_2}{H, \rho, A \models e_1 = e_2} \quad \text{EA-NEqual} \\
\\
\frac{H, \rho \vdash x \Downarrow o \quad (o, f) \in A}{H, \rho, A \models \mathbf{acc}(x.f)} \quad \text{EA-Acc} \\
\\
\frac{A_1 = A \setminus A_2 \quad H, \rho, A_1 \models \phi_1 \quad H, \rho, A_2 \models \phi_2}{H, \rho, A \models \phi_1 * \phi_2} \quad \text{EA-SepOp}
\end{array}$$

### 3.3 Footprint ( $\text{footprint}_{H,\rho}(\phi) = A_d$ )

$$\begin{aligned}
\text{footprint}_{H,\rho}(\text{true}) &= \emptyset \\
\text{footprint}_{H,\rho}(e_1 = e_2) &= \emptyset \\
\text{footprint}_{H,\rho}(e_1 \neq e_2) &= \emptyset \\
\text{footprint}_{H,\rho}(\text{acc}(e.f)) &= \{(o, f)\} \text{ where } H, \rho \vdash e \Downarrow o \\
\text{footprint}_{H,\rho}(\phi_1 * \phi_2) &= \text{footprint}_{H,\rho}(\phi_1) \cup \text{footprint}_{H,\rho}(\phi_2)
\end{aligned}$$

### 3.4 Small-step ( $((H, S) \rightarrow (H, S))$ )

$$\frac{H, \rho \vdash x \Downarrow o \quad H, \rho \vdash y \Downarrow v_y \quad (o, f) \in A \quad H' = H[o \mapsto [f \mapsto v_y]]}{(H, (\rho, A, x.f := y; \bar{s}) \cdot S) \rightarrow (H', (\rho, A, \bar{s}) \cdot S)} \text{ESFIELDASSIGN}$$

$$\frac{H, \rho \vdash e \Downarrow v \quad \rho' = \rho[x \mapsto v]}{(H, (\rho, A, x := e; \bar{s}) \cdot S) \rightarrow (H, (\rho', A, \bar{s}) \cdot S)} \text{ESVARASSIGN}$$

$$\frac{\rho' = \rho[x \mapsto o] \quad H(o) = \perp \quad \text{fields}(C) = f \quad A' = A * (o, f_i) \quad H' = H[o \mapsto [((f_i, \text{null}))]]}{(H, (\rho, A, x := \text{new } C; \bar{s}) \cdot S) \rightarrow (H', (\rho', A', \bar{s}) \cdot S)} \text{ESNEWOBJ}$$

$$\frac{H, \rho \vdash x \Downarrow v_x \quad \rho' = \rho[\text{result} \mapsto v_x]}{(H, (\rho, a, \text{return } x; \bar{s}) \cdot S) \rightarrow (H, (\rho', a, \bar{s}) \cdot S)} \text{ESRETURN}$$

$$\frac{
\begin{array}{l}
H, \rho \vdash y \Downarrow o \quad H, \rho \vdash z \Downarrow v \quad H(o) = (C, fvf) \\
\text{mbody}(C, m) = \bar{r} \quad \text{mparam}(C, m) = (T, w) \quad \text{mpre}(C, m) = \phi \\
\rho' = [\text{this} \mapsto o, w \mapsto v] \quad H, \rho', A \models \phi \quad A' = \text{footprint}_{H,\rho'}(\phi)
\end{array}
}{(H, (\rho, A, x := y.m(z); \bar{s}) \cdot S) \rightarrow (H, (\rho', A', \bar{r}) * (\rho, A \setminus A', x := y.m(z); \bar{s}) \cdot S)} \text{ESAPP}$$

$$\frac{
\begin{array}{l}
\text{mpost}(C, m) = \phi \\
H, \rho', A' \models \phi \quad A'' = \text{footprint}_{H,\rho'}(\phi) \quad H, \rho' \vdash \text{result} \Downarrow v_r
\end{array}
}{(H, (\rho', A', \emptyset) * (\rho, A, x := y.m(z); \bar{s}) \cdot S) \rightarrow (H, (\rho[x \mapsto v_r], A * A'', \bar{s}) \cdot S)} \text{ESAPPFINISH}$$

$$\frac{H, \rho, A \models \phi}{(H, (\rho, A, \text{assert } \phi; \bar{s}) \cdot S) \rightarrow (H, (\rho, A, \bar{s}) \cdot S)} \text{ESASSERT}$$

$$\frac{H, \rho, A \models \phi \quad A' = A \setminus \text{footprint}_{H, \rho}(\phi)}{(H, (\rho, A, \text{release } \phi; \bar{s}) \cdot S) \rightarrow (H, (\rho, A', \bar{s}) \cdot S)} \text{ESRELEASE}$$

## 4 Theorems

Hoare preserves self-framing

$$\begin{aligned} \forall \Gamma, \phi_1, \phi_2, s : \Gamma \vdash \{\phi_1\} s \{\phi_2\} \\ \implies \text{static-footprint}(\phi_1) \vdash_{\text{sfrm}} \phi_1 \\ \implies \text{static-footprint}(\phi_2) \vdash_{\text{sfrm}} \phi_2 \end{aligned}$$

Hoare progress

$$\begin{aligned} \forall \Gamma, \phi_1, \phi_2, s, H_1, \rho_1, A_1 : \Gamma \vdash \{\phi_1\} s \{\phi_2\} \\ \implies H_1, \rho_1, A_1 \models \phi_1 \\ \implies \exists H_2, \rho_2, A_2 : (H_1, (\rho_1, A_1, s'; \bar{s}) \cdot S) \rightarrow^* (H_2, (\rho_2, A_2, \bar{s}) \cdot S) \end{aligned}$$

Hoare preservation

$$\begin{aligned} \forall \Gamma, \phi_1, \phi_2, s, H_1, H_2, \rho_1, \rho_2, A_1, A_2 : \Gamma \vdash \{\phi_1\} s \{\phi_2\} \\ \implies H_1, \rho_1, A_1 \models \phi_1 \\ \implies (H_1, (\rho_1, A_1, s'; \bar{s}) \cdot S) \rightarrow^* (H_2, (\rho_2, A_2, \bar{s}) \cdot S) \\ \implies H_2, \rho_2, A_2 \models \phi_2 \end{aligned}$$