Axiomatic Semantics for Java$_{\ell ight}$

David von Oheimb, TU München

Project Bali

http://isabelle.in.tum.de/Bali/

Axiomatic Semantics for Java$_{\ell ight}$

Hoare Logic for IMP$_P$ (FST&TCS'99)

- Local Variables
- Mutual Recursion
- Auxiliary Variables
- MGF approach

Application to Java$_{\ell ight}$ (work in progress)

- Exceptions
- Expressions
- Dynamic Method Lookup
- Class Initialization

Abstract Syntax of IMP$_P$

$$
\begin{array}{rll}
com & = & \mathtt{SKIP} \mid com;\ com \mid vname\ \mathtt{:=}\ aexp \\
& \mid & \mathtt{IF}\ bexp\ \mathtt{THEN}\ com\ \mathtt{ELSE}\ com \mid \mathtt{WHILE}\ bexp\ \mathtt{DO}\ com \\
& \mid & \mathtt{LOCAL}\ loc\ \mathtt{:=}\ aexp\ \mathtt{IN}\ com \\
& \mid & vname\ \mathtt{:=}\ \mathtt{CALL}\ pname(aexp) \mid \mathtt{Call}\ pname
\end{array}
$$

$$aexp = state \to val$$

$$vname = \mathsf{Glb}\ glb \mid \mathsf{Loc}\ loc$$

body :: $pname \to com$ finitely many procedures!

Operational Semantics for IMP$_P$

$$Local\ \frac{\langle c,\ \sigma_0[a\ \sigma_0/X]\rangle \longrightarrow \sigma_1}{\langle \texttt{LOCAL}\ X\ \texttt{:=}\ a\ \texttt{IN}\ c,\ \sigma_0\rangle \longrightarrow \sigma_1[\sigma_0\langle X\rangle/X]}$$

$$CALL\ \frac{\langle \texttt{Call}\ p,\ (\mathsf{setlocs}\ \sigma_0\ \mathsf{newlocs})[a\ \sigma_0/\texttt{Arg}]\rangle \longrightarrow \sigma_1}{\langle X\texttt{:=CALL}\ p(a),\sigma_0\rangle \longrightarrow (\mathsf{setlocs}\ \sigma_1\ (\mathsf{getlocs}\ \sigma_0))[X\texttt{:=}\sigma_1\langle\texttt{Res}\rangle]}$$

$$Call\ \frac{\langle \mathsf{body}\ p,\ \sigma_0\rangle \longrightarrow \sigma_1}{\langle \texttt{Call}\ p,\ \sigma_0\rangle \longrightarrow \sigma_1}$$

Extensions to Hoare Triples

Assertions (first approach) $assn = state \to bool$

Auxiliary Variables $\vdash \{X{=}Z\}\ c\ \{X{=}Z\}$

intended meaning: $\vdash \forall Z.\ \{\lambda\sigma.\ \sigma\langle X\rangle{=}Z\}\ c\ \{\lambda\sigma.\ \sigma\langle X\rangle{=}Z\}$

Assumptions $\{\{P\}\ \texttt{Call}\ F\ \{Q\}\} \vdash \{P\}$ body $F\ \{Q\}$

Assertions (revised version) $\alpha\ assn = \alpha \to state \to bool$

Hoare Triples $\{\alpha\ assn\}\ com\ \{\alpha\ assn\}$

Validity (wrt. Partial Correctness, preliminary definition)

$$\models \{P\}\ c\ \{Q\} \equiv\ \forall Z\ \sigma\ \sigma'.\ P\ Z\ \sigma \Longrightarrow \langle c,\sigma\rangle \longrightarrow \sigma' \Longrightarrow Q\ Z\ \sigma'$$

Non-Procedure Rules

Preliminary Judgments: $\vdash \_ :: \alpha\ triple \to bool$

$$Local\ \frac{\vdash \{P\}\ c\ \{\lambda Z\,\sigma.\ Q\ Z\,(\sigma[\sigma_0\ \langle X\rangle/X])\}}{\vdash \{\lambda Z\,\sigma.\ \sigma_0{=}\sigma \wedge P\ Z\,(\sigma[a\,\sigma/X])\}\ \texttt{LOCAL}\ X\texttt{:=}a\ \texttt{IN}\ c\ \{Q\}}$$

$$Local\ \frac{\vdash \{P\}\ c\ \{Q[old\ X/X]\}}{\vdash \{P[a/X]\}\ \texttt{LOCAL}\ X\texttt{:=}a\ \texttt{IN}\ c\ \{Q\}}$$

$$conseq\ \frac{\begin{array}{c}\forall Z\,\sigma.\ P\ Z\,\sigma \Longrightarrow \exists P'\ Q'.\ \vdash \{P'\}\ c\ \{Q'\} \wedge \\ \forall \sigma'.\ (\forall Z'.\ P'\ Z'\ \sigma \Longrightarrow Q'\ Z'\ \sigma') \Longrightarrow Q\ Z\,\sigma'\end{array}}{\vdash \{P\}\ c\ \{Q\}}$$

$$disj\ \frac{\vdash \{P\}\ c\ \{Q\} \qquad \vdash \{P'\}\ c\ \{Q'\}}{\vdash \{P \vee P'\}\ c\ \{Q \vee Q'\}}$$

Nested Call Rule

Revised Judgments: $\_ \vdash \_ :: \alpha\ triple\ set \to \alpha\ triple \to bool$

$$CallN\ \frac{\{\{P\}\ \texttt{Call}\ p\ \{Q\}\} \cup \Gamma\ \vdash\ \{P\}\ \mathsf{body}\ p\ \{Q\}}{\Gamma \vdash \{P\}\ \texttt{Call}\ p\ \{Q\}} \qquad asm\ \frac{t \in \Gamma}{\Gamma \vdash t}$$

Example: P calls Q and R, Q calls R, and R calls P and Q:

$$\dfrac{\vdots\ \{P\} \vdash (\text{body of } P)\ \vdots}{\emptyset \vdash \texttt{Call}\ P}$$

where

$$\dfrac{\vdots\ \{P,Q\} \vdash (\text{body of } Q)\ \vdots}{\{P\} \vdash \texttt{Call}\ Q} \quad \text{with} \quad \dfrac{\vdots\ \{P,Q,R\} \vdash (\text{body of } R)\ \vdots}{\{P,Q\} \vdash \texttt{Call}\ R} \quad \text{from} \quad \dfrac{}{\{P,Q,R\} \vdash \texttt{Call}\ P} \quad \dfrac{}{\{P,Q,R\} \vdash \texttt{Call}\ Q}$$

$$\dfrac{\vdots\ \{P,R\} \vdash (\text{body of } R)\ \vdots}{\{P\} \vdash \texttt{Call}\ R} \quad \text{from} \quad \dfrac{}{\{P,R\} \vdash \texttt{Call}\ P} \quad \dfrac{\vdots\ \{P,Q,R\} \vdash (\text{body of } Q)\ \vdots}{\{P,R\} \vdash \texttt{Call}\ Q} \quad \text{from} \quad \dfrac{}{\{P,Q,R\} \vdash \texttt{Call}\ R}$$

Simultaneous Call Rule

Final Judgments: $\_ \Vdash \_ :: \alpha\ triple\ set \to \alpha\ triple\ set \to bool$

$$Call\ \frac{\Gamma \cup \{\{P_i\}\texttt{Call}\ i\{Q_i\} \mid i \in ps\} \Vdash \{\{P_i\}\mathsf{body}\ i\{Q_i\} \mid i \in ps\}}{\Gamma \vdash \{\{P_i\}\ \texttt{Call}\ i\{Q_i\}\} \mid i \in ps\}}$$

$$empty\ \frac{}{\Gamma \Vdash \emptyset} \qquad insert\ \frac{\Gamma \vdash t \quad \Gamma \Vdash ts}{\Gamma \Vdash \{t\} \cup ts} \qquad subset\ \frac{\Gamma \Vdash ts' \quad ts \subseteq ts'}{\Gamma \Vdash ts}$$

Example:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\{P,Q,R\} \vdash \texttt{Call}\ P} \qquad \dfrac{}{\{P,Q,R\} \vdash \texttt{Call}\ Q} \qquad \dfrac{}{\{P,Q,R\} \vdash \texttt{Call}\ R}}{\vdots\ \{P,Q,R\} \Vdash (\text{bodies of } P,\ Q \text{ and } R)\ \vdots}}{\emptyset \Vdash \{\texttt{Call}\ P,\ \texttt{Call}\ Q,\ \texttt{Call}\ R\}}}{\emptyset \vdash \texttt{Call}\ P}$$

Soundness Proof: Validity Revisited

Soundness: $\emptyset \vdash t \Longrightarrow \emptyset \models t$

Proof: $\Gamma \Vdash ts \Longrightarrow \Gamma \models\!\!\!\!\models ts$ by rule induction

Recursive Depth: an annotation

$$Call\ \frac{\langle \mathsf{body}\ p,\ \sigma_0 \rangle -n\rightarrow\ \sigma_1}{\langle \mathtt{CALL}\ p,\ \sigma_0 \rangle -n{+}1\rightarrow\ \sigma_1}$$

$$\models n{:}\{P\}\ c\ \{Q\} \equiv \forall Z\ \sigma\ \sigma'.\ P\ Z\ \sigma \Longrightarrow \langle c,\sigma\rangle -n\rightarrow\ \sigma' \Longrightarrow\ Q\ Z\ \sigma'$$

$$\models\!\!\!\!\models n{:}ts \equiv \forall t \in ts.\ \models n{:}t$$

$$\Gamma \models\!\!\!\!\models ts \equiv \forall n.\ \models\!\!\!\!\models n{:}\Gamma \Longrightarrow\ \models\!\!\!\!\models n{:}ts$$

(Relative) Completeness Proof: MGF Approach

$\texttt{MGT}\ c \equiv \{\lambda Z\, \sigma_0.\ Z = \sigma_0\}\ c\ \{\lambda Z\, \sigma_1.\ \langle c, Z\rangle \longrightarrow \sigma_1\}$

Completeness Lemma: $\emptyset \vdash \texttt{MGT}\ c \Longrightarrow \emptyset {\models} \{P\}\ c\ \{Q\} \Longrightarrow \emptyset \vdash \{P\}\ c\ \{Q\}$

Main Proof: $\emptyset \vdash \texttt{MGT}\ c$ (by structural/rule induction)

Version 1: Nested Structural Induction

MGT-lemma: $(\forall p.\ \Gamma \vdash \texttt{MGT}\ (\texttt{Call}\ p)) \Longrightarrow \Gamma \vdash \texttt{MGT}\ c$ (by Struct. Ind.)

Auxiliary induction on the number of procedures (strengthened goal),

using 2×MGT-lemma, $CallN$, asm, and finiteness

Version 2: Simultaneous Structural Induction

using 1×MGT-lemma, $Call$, $empty$, $insert$, asm, and finiteness

(Relative) Completeness Proof (cont'd)

Version 3: Rule Induction for proving

$$\forall \sigma_0\, \sigma_1.\ \langle c, \sigma_0 \rangle \longrightarrow \sigma_1 \Longrightarrow \emptyset \vdash \{\lambda Z\, \sigma.\ \sigma = \sigma_0\}\ c\ \{\lambda Z\, \sigma.\ \sigma = \sigma_1\}$$

using a trivial version of $Call$ and an unfolding variant of the $Loop$ rule:

$$LoopT\ \frac{\Gamma \vdash \{P \wedge b\}\ c\ \{Q\} \qquad \Gamma \vdash \{Q\}\ \texttt{WHILE}\ b\ \texttt{DO}\ c\ \{R\}}{\Gamma \vdash \{P \wedge b\}\ \texttt{WHILE}\ b\ \texttt{DO}\ c\ \{R\}}$$

Main proof requires $diverg\ \dfrac{}{\Gamma \vdash \{\lambda Z\, \sigma.\ \neg\exists \sigma'.\ \langle c, \sigma \rangle \longrightarrow \sigma'\}\ c\ \{Q\}}$

+ simpler proof
+ does not require assumptions nor auxiliary variables nor finiteness
– requires two additional rules
– rather operational style (okay for meta theory only)

Application to Java$_{\ell ight}$, Special Features

Expressions

Side-effecting expressions should not be flattened to statements.

New Assertion Format: $\alpha\ assn = \alpha \to res\ list \times state \to bool$

$$Super\ \frac{}{G,\Gamma \vdash \{\text{Normal}\ P\}\ \text{Super}\ \{\lambda Z\,(Y,\sigma).\ \exists Y'.\ Y = (\text{the}\ (\text{locals}\ \sigma\ \text{This}))\#Y' \wedge P\ Z\,(Y',\sigma)\}}$$

Exceptions

Direct representation within assertions,

e.g. Normal $P \equiv \lambda Z\,(Y,\sigma).\ P\ Z\,(Y,\sigma) \wedge \text{fst}\ \sigma = \text{None}$

$$Xcpt\ \frac{}{G,\Gamma \vdash \{\lambda Z\,\sigma.\ P\ Z\,\sigma \wedge \text{fst}\ \sigma \neq \text{None}\}\ e\ \{\text{arbitrary}\div\!\!\to P\}}$$

Dynamic Method Lookup

Statements depend on previous results, i.e. on the state.

Idea: Make case distinction over all possible values (finitely many).

$$\frac{G,\Gamma \vdash \{\mathsf{Normal}\ P\}\ e\ \{P'\} \qquad G,\Gamma \vdash \{\{\lambda Z\,(Y,\sigma).\ \mathsf{hd}\ Y{=}b \wedge P'\,Z\,(Y,\sigma)\}\ (\mathsf{if}\ b\ \mathsf{then}\ c_1\ \mathsf{else}\ c_2)\ \{Q\} \mid b \in bool\}}{G,\Gamma \vdash \{\mathsf{Normal}\ P\}\ \mathsf{If}(e)\ c_1\ \mathsf{Else}\ c_2\ \{Q\}}$$

Class Initialization

Mutual “procedure” calls (without real recursion) again

Soundness & Completeness Proofs

Prototypical versions

Conclusion

Axiomatic Semantics for Java is inherently complex.

- Extensions to classical Hoare calculus seem unavoidable.
- Completeness proof is particularly challenging.

- Theorem prover enforces fully formal treatment.
- Theorem prover assists with all the details.

Axiomatic Semantics for Java is at least feasible.