An Easy Exercise in Hoare's Logic: Imperative Expressions

Andrzej Tarlecki

Institute of Informatics, University of Warsaw

and

Institute of Computer Science, Polish Academy of Sciences

Warsaw, Poland

Background

- student's question:

  Where does one put a loop invariant when the loop guard may have side effects?

- my answer:

  Hoare logic for while-programs with imperative expressions: this **must** *be well-known!*

- search through the literature, asking THOSE-WHO-MUST-KNOW: **:-(**

- one way to proceed:

  D.I.Y.

. . . and present it to WG1.3 to find out who and where did this earlier. . .

The standard case

Built-in data type:

- signature: $\Sigma_{\mathbf{D}} = \langle\langle\Omega_n\rangle_{n\in N}, \langle\Pi_n\rangle_{n\in N}\rangle$
- semantic structure $\mathbf{D}$ with:
  - carrier D
  - operations $f_{\mathbf{D}}\colon D^n \to D$ for $f \in \Omega_n$, $n \in N$
  - predicates $p_{\mathbf{D}}\colon D^n \to \mathbf{B}$ for $p \in \Pi_n$, $n \in N$

  where $\mathbf{B} = \{\mathbf{ff}, \mathbf{tt}\}$

Single sorted...
only to keep the notation simple...

While-programs

$$x \in \mathbf{Var} ::= \cdots$$

$$e \in \mathbf{Exp} ::= x \mid f(e_1, \ldots, e_n) \quad (\text{for each } f \in \Omega_n,\ n \in N)$$

$$b \in \mathbf{BExp} ::= \mathsf{true} \mid \mathsf{false} \mid \neg b' \mid b_1 \wedge b_2 \mid p(e_1, \ldots, e_n) \quad (\text{for each } p \in \Pi_n,\ n \in N)$$

$$S \in \mathbf{Stmt} ::= x := e \mid \mathsf{skip} \mid S_1; S_2 \mid \mathsf{if}\ b\ \mathsf{then}\ S_1\ \mathsf{else}\ S_2 \mid \mathsf{while}\ b\ \mathsf{do}\ S'$$

Semantics

$$\mathcal{E} : \mathbf{Exp} \to \mathbf{State} \to D$$

$$\mathcal{B} : \mathbf{BExp} \to \mathbf{State} \to \mathbf{B}$$

$$\mathcal{S} : \mathbf{Stmt} \to \mathbf{State} \to \mathbf{State}$$

where $\mathbf{State} = \mathbf{Var} \to D$.

Standard semantic clauses omitted

Judgements

Whenever program $S \in \mathbf{Stmt}$ *starts in a state satisfying precondtion* $\varphi \in \mathcal{L}$ *and terminates successfully, then the final state satisfies postcondition* $\psi \in \mathcal{L}$

$$\{\varphi\}\, S\, \{\psi\}$$

Semantic satisfaction

$$\models \{\varphi\}\, S\, \{\psi\}$$

iff

for $s \in \mathbf{State}$, if $[\![\varphi]\!]\, s = \mathbf{tt}$ and $\mathcal{S}[\![S]\!]\, s = s'$ then $[\![\psi]\!]\, s' = \mathbf{tt}$

where $[\![\_]\!] : \mathcal{L} \to \mathbf{State} \to \mathbf{B}$ gives the semantics of formulae in $\mathcal{L}$

Proof system

$$\frac{}{\{\varphi[x \mapsto e]\}\; x := e\; \{\varphi\}}$$

$$\frac{}{\{\varphi\}\; \mathsf{skip}\; \{\varphi\}}$$

$$\frac{\{\varphi\}\; S_1\; \{\theta\} \qquad \{\theta\}\; S_2\; \{\psi\}}{\{\varphi\}\; S_1; S_2\; \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \qquad \{\varphi\}\; S\; \{\psi\} \qquad \psi \Rightarrow \psi'}{\{\varphi'\}\; S\; \{\psi'\}}$$

$$\frac{\{\varphi \wedge b\}\; S_1\; \{\psi\} \qquad \{\varphi \wedge \neg b\}\; S_2\; \{\psi\}}{\{\varphi\}\; \mathsf{if}\ b\ \mathsf{then}\ S_1\ \mathsf{else}\ S_2\; \{\psi\}}$$

$$\frac{\{\varphi \wedge b\}\; S\; \{\varphi\}}{\{\varphi\}\; \mathsf{while}\ b\ \mathsf{do}\ S\; \{\varphi \wedge \neg b\}}$$

Soundness

if $Th(\mathcal{L}) \vdash \{\varphi\}\, S\, \{\psi\}$ then $\models \{\varphi\}\, S\, \{\psi\}$

where $Th(\mathcal{L})$ is the $\mathcal{L}$-theory of $\mathbf{D}$

Completeness

If $\mathcal{L}$ is expressive in $\mathbf{D}$ for $\mathbf{Stmt}$ then:

if $\models \{\varphi\}\, S\, \{\psi\}$ then $Th(\mathcal{L}) \vdash \{\varphi\}\, S\, \{\psi\}$

While-programs with imperative expressions

$$x \in \mathbf{Var} ::= \cdots$$

$$E \in \mathbf{IExp} ::= x \mid f(E_1, \ldots, E_n) \quad (\text{for each } f \in \Omega_n,\ n \in N)$$
$$\mid E' \text{ after } S$$

$$B \in \mathbf{IBExp} ::= \text{true} \mid \text{false} \mid \neg B' \mid B_1 \wedge B_2 \mid p(E_1, \ldots, E_n) \quad (\text{for each } p \in \Pi_n,\ n \in N)$$
$$\mid B' \text{ after } S$$

$$S \in \mathbf{Stmt} ::= x := E \mid \text{skip} \mid S_1; S_2 \mid \text{if } B \text{ then } S_1 \text{ else } S_2 \mid \text{while } B \text{ do } S'$$

Semantics

$$\mathcal{E}_\mathbf{I} : \mathbf{Exp} \to \mathbf{State} \to D \times \mathbf{State}$$
$$\mathcal{B}_\mathbf{I} : \mathbf{BExp} \to \mathbf{State} \to \mathbf{B} \times \mathbf{State}$$
$$\mathcal{S} : \mathbf{Stmt} \to \mathbf{State} \to \mathbf{State}$$

Expected semantic clauses omitted

Judgements

$\{\varphi\}\, S\, \{\psi\}$

Whenever computation $S \in \mathbf{Stmt}$ *starts in a state satisfying* φ *and terminates, the final state satisfies* $\psi \in \mathcal{L}$

$\{\varphi\}\, E\, \{\lambda v{:}\mathbf{D} \cdot \psi\}$

Whenever computation of $E \in \mathbf{IExp}$ *starts in a state satisfying* φ *and terminates, the final state satisfies* $\psi[v \mapsto d]$, *where* $d \in D$ *is the value of* E

$\{\varphi\}\, B\, \{\lambda v{:}\mathbf{B} \cdot \psi\}$

Whenever computation of $B \in \mathbf{IBExp}$ *starts in a state satisfying* φ *and terminates, the final state satisfies* $\psi[v \mapsto b]$, *where* $b \in \mathbf{B}$ *is the value of* B

Semantic satisfaction

$\models \{\varphi\}\, S\, \{\psi\}$

for $s \in \mathbf{State}$, *if* $[\![\varphi]\!]\, s = \mathbf{tt}$ *and* $\mathcal{S}[\![S]\!]\, s = s'$ *then* $[\![\psi]\!]\, s' = \mathbf{tt}$

$\models \{\varphi\}\, E\, \{\lambda v{:}\mathbf{D} \cdot \psi\}$

for $s \in \mathbf{State}$, *if* $[\![\varphi]\!]\, s = \mathbf{tt}$ *and* $\mathcal{E}_{\mathbf{I}}[\![E]\!]\, s = \langle d, s' \rangle$ *then* $[\![\psi]\!]\, (s'[v \mapsto d]) = \mathbf{tt}$

$\models \{\varphi\}\, B\, \{\lambda v{:}\mathbf{B} \cdot \psi\}$

for $s \in \mathbf{State}$, *if* $[\![\varphi]\!]\, s = \mathbf{tt}$ *and* $\mathcal{B}_{\mathbf{I}}[\![B]\!]\, s = \langle b, s' \rangle$ *then* $[\![\psi]\!]\, (s'[v \mapsto b]) = \mathbf{tt}$

Proof system

$$\{\varphi\}\ S\ \{\psi\}$$

$$\{\varphi\}\ E\ \{\lambda v{:}\mathbf{D} \cdot \psi\}$$

$$\{\varphi\}\ B\ \{\lambda v{:}\mathbf{B} \cdot \psi\}$$

$$\{\varphi\}\ S\ \{\psi\}$$

$$\frac{\{\varphi\}\ E\ \{\lambda v{:}\mathbf{D}\cdot\psi\}}{\{\varphi\}\ x := E\ \{\psi[v \mapsto x]\}}$$

$$\frac{}{\{\varphi\}\ \mathsf{skip}\ \{\varphi\}}$$

$$\frac{\{\varphi\}\ S_1\ \{\theta\} \quad \{\theta\}\ S_2\ \{\psi\}}{\{\varphi\}\ S_1; S_2\ \{\psi\}}$$

$$\frac{\varphi' \Rightarrow \varphi \quad \{\varphi\}\ S\ \{\psi\} \quad \psi \Rightarrow \psi'}{\{\varphi'\}\ S\ \{\psi'\}}$$

$$\frac{\{\varphi\}\ B\ \{\lambda v{:}\mathbf{B}\cdot\theta\} \quad \{\theta[v \mapsto \mathsf{true}]\}\ S_1\ \{\psi\} \quad \{\theta[v \mapsto \mathsf{false}]\}\ S_2\ \{\psi\}}{\{\varphi\}\ \mathsf{if}\ B\ \mathsf{then}\ S_1\ \mathsf{else}\ S_2\ \{\psi\}}$$

$$\frac{\{\varphi\}\ B\ \{\lambda v{:}\mathbf{B}\cdot\psi\} \quad \{\psi[v \mapsto \mathsf{true}]\}\ S\ \{\varphi\}}{\{\varphi\}\ \mathsf{while}\ B\ \mathsf{do}\ S\ \{\psi[v \mapsto \mathsf{false}]\}}$$

$$\{\varphi\}\ E\ \{\lambda v{:}\mathbf{D}\cdot\psi\}$$

$$\frac{\varphi'\Rightarrow\varphi \qquad \{\varphi\}\ E\ \{\lambda v{:}\mathbf{D}\cdot\psi\} \qquad \psi\Rightarrow\psi'}{\{\varphi'\}\ E\ \{\lambda v{:}\mathbf{D}\cdot\psi'\}}$$

$$\frac{\{\varphi\}\ S\ \{\theta\} \qquad \{\theta\}\ E\ \{\lambda v{:}\mathbf{D}\cdot\psi\}}{\{\varphi\}\ E\ \textsf{after}\ S\ \{\lambda v{:}\mathbf{D}\cdot\psi\}}$$

$$\frac{}{\{\psi[v\mapsto x]\}\ x\ \{\lambda v{:}\mathbf{D}\cdot\psi\}}$$

$$\frac{}{\{\psi[v\mapsto f()]\}\ f()\ \{\lambda v{:}\mathbf{D}\cdot\psi\}}$$

$$\frac{\begin{array}{c}\{\varphi\}\ E_1\ \{\lambda v{:}\mathbf{D}\cdot\theta_1\}\\ \{\theta_1[v\mapsto v_1]\}\ E_2\ \{\lambda v{:}\mathbf{D}\cdot\theta_2\} \quad \ldots \quad \{\theta_{n-1}[v\mapsto v_{n-1}]\}\ E_n\ \{\lambda v{:}\mathbf{D}\cdot\theta_n\}\\ \theta_n[v\mapsto v_n]\Rightarrow\psi[v\mapsto f(v_1,\ldots,v_n)]\end{array}}{\{\varphi\}\ f(E_1,\ldots,E_n)\ \{\lambda v{:}\mathbf{D}\cdot\psi\}}$$

$$\{\varphi\}\ B\ \{\lambda v{:}\mathbf{B}\cdot\psi\}$$

$$\frac{\varphi'\Rightarrow\varphi \quad \{\varphi\}\ B\ \{\lambda v{:}\mathbf{B}\cdot\psi\} \quad \psi\Rightarrow\psi'}{\{\varphi'\}\ B\ \{\lambda v{:}\mathbf{B}\cdot\psi'\}}$$

$$\frac{\{\varphi\}\ S\ \{\theta\} \quad \{\theta\}\ B\ \{\lambda v{:}\mathbf{B}\cdot\psi\}}{\{\varphi\}\ B \text{ after } S\ \{\lambda v{:}\mathbf{B}\cdot\psi\}}$$

$$\frac{\{\varphi\}\ B\ \{\lambda v{:}\mathbf{B}\cdot\psi\}}{\{\varphi\}\ \neg B\ \{\lambda v{:}\mathbf{B}\cdot\psi[v\mapsto\neg v]\}}$$

$$\frac{}{\{\psi[v\mapsto p()]\}\ p()\ \{\lambda v{:}\mathbf{B}\cdot\psi\}}$$

$$\dots$$

$$\frac{\begin{array}{c}\{\varphi\}\ E_1\ \{\lambda v{:}\mathbf{D}\cdot\theta_1\} \\ \{\theta_1[v\mapsto v_1]\}\ E_2\ \{\lambda v{:}\mathbf{D}\cdot\theta_2\} \quad \dots \quad \{\theta_{n-1}[v\mapsto v_{n-1}]\}\ E_n\ \{\lambda v{:}\mathbf{D}\cdot\theta_n\} \\ \theta_n[v\mapsto v_n]\Rightarrow\psi[v\mapsto p(v_1,\dots,v_n)]\end{array}}{\{\varphi\}\ p(E_1,\dots,E_n)\ \{\lambda v{:}\mathbf{B}\cdot\psi\}}$$

Soundness

if $Th(\mathcal{L}) \vdash \{\varphi\}\, S\, \{\psi\}$ then $\models \{\varphi\}\, S\, \{\psi\}$

if $Th(\mathcal{L}) \vdash \{\varphi\}\, E\, \{\lambda v{:}\mathbf{D} \cdot \psi\}$ then $\models \{\varphi\}\, E\, \{\lambda v{:}\mathbf{D} \cdot \psi\}$

if $Th(\mathcal{L}) \vdash \{\varphi\}\, B\, \{\lambda v{:}\mathbf{B} \cdot \psi\}$ then $\models \{\varphi\}\, B\, \{\lambda v{:}\mathbf{B} \cdot \psi\}$

Completeness

If $\mathcal{L}$ is expressive in $\mathbf{D}$ for $\mathbf{Stmt}$ then:

if $\models \{\varphi\}\, S\, \{\psi\}$ then $Th(\mathcal{L}) \vdash \{\varphi\}\, S\, \{\psi\}$

if $\models \{\varphi\}\, E\, \{\lambda v{:}\mathbf{D} \cdot \psi\}$ then $Th(\mathcal{L}) \vdash \{\varphi\}\, E\, \{\lambda v{:}\mathbf{D} \cdot \psi\}$

if $\models \{\varphi\}\, B\, \{\lambda v{:}\mathbf{B} \cdot \psi\}$ then $Th(\mathcal{L}) \vdash \{\varphi\}\, B\, \{\lambda v{:}\mathbf{B} \cdot \psi\}$

Further comments

- other imperative constructs (in particular: recursion)
- backward vs. forward reasoning style
- wp-reasoning, binary conditions, dynamic/algorithmic logic, ...
- exact assumption about the underlying logic $\mathcal{L}$
- exact assumptions on the built-in data type, and on the use of new variables
- an institution-independent version **:-)**

is this of any interest?