

Proving things

Some meta-properties (mentioned or not explicitly so far):

- semantics of expressions: *free variables*, *referential transparency*
- operational semantics: *determinism*, *computations compose*
- natural semantics: *determinism*, $\leadsto = \Rightarrow^*$
- denotational semantics: *adequacy w.r.t. computations*
- Hoare's logic: *soundness*, *completeness*
- total correctness: *soundness*, *completeness*
- ...

Proof methods:

- *structural induction*
- *induction on the length of computations*
- *induction on the derivation trees*
- *fixed-point induction*
- ...

Sample proofs follow;
semantics runs the show!

Structural induction for expressions

$$e ::= N \mid x \mid e_1 + e_2 \mid e_1 * e_2 \mid e_1 - e_2$$

Given a property $P(-)$ of expressions:

IF

- $P(N)$, for all $N \in \mathbf{Num}$
- $P(x)$, for all $x \in \mathbf{Var}$
- $P(e_1 + e_2)$ follows from $P(e_1)$ and $P(e_2)$, for all $e_1, e_2 \in \mathbf{Exp}$
- $P(e_1 * e_2)$ follows from $P(e_1)$ and $P(e_2)$, for all $e_1, e_2 \in \mathbf{Exp}$
- $P(e_1 - e_2)$ follows from $P(e_1)$ and $P(e_2)$, for all $e_1, e_2 \in \mathbf{Exp}$

THEN

- $P(e)$ for all $e \in \mathbf{Exp}$.

Inductive definitions

Free variables in expressions $FV(e) \subset \mathbf{Var}$:

$$FV(N) = \emptyset$$

$$FV(x) = \{x\}$$

$$FV(e_1 + e_2) = FV(e_1) \cup FV(e_2)$$

$$FV(e_1 * e_2) = FV(e_1) \cup FV(e_2)$$

$$FV(e_1 - e_2) = FV(e_1) \cup FV(e_2)$$

Fact: For each expression $e \in \mathbf{Exp}$, the set $FV(e)$ of its free variables is finite.

Proof: by structural induction (easy)

Fact: *The meaning of expression depends only on the valuation of its free variables:*
for any $e \in \mathbf{Exp}$ and $s, s' \in \mathbf{State}$

if $s x = s' x$ for all $x \in FV(e)$ then $\mathcal{E}[e] s = \mathcal{E}[e] s'$

Proof: (by structural induction)

- for $N \in \mathbf{Num}$,
 $\mathcal{E}[N] s = \mathcal{N}[N] = \mathcal{E}[N] s'$
- for $x \in \mathbf{Var}$,
 $\mathcal{E}[x] s = s x = s' x = \mathcal{E}[x] s'$
- for $e_1, e_2 \in \mathbf{Exp}$,
 $\mathcal{E}[e_1 + e_2] s = \mathcal{E}[e_1] s + \mathcal{E}[e_2] s = \mathcal{E}[e_1] s' + \mathcal{E}[e_2] s' = \mathcal{E}[e_1 + e_2] s'$
- ...

by the inductive hypothesis,
since $FV(e_1), FV(e_2) \subseteq FV(e_1 + e_2)$

Referential transparency

Substitution of e' for x in e results in $e[e'/x]$:

$$\begin{aligned} N[e'/x] &= N \\ x'[e'/x] &= \begin{cases} e' & \text{if } x = x' \\ x' & \text{if } x \neq x' \end{cases} \end{aligned}$$

$$(e_1 + e_2)[e'/x] = e_1[e'/x] + e_2[e'/x]$$

$$(e_1 * e_2)[e'/x] = e_1[e'/x] * e_2[e'/x]$$

$$(e_1 - e_2)[e'/x] = e_1[e'/x] - e_2[e'/x]$$

Then:

$$\mathcal{E}[e[e'/x]] s = \mathcal{E}[e] s[x \mapsto \mathcal{E}[e'] s]$$

Proof: by structural induction (easy)

Operational semantics: computations compose

Fact: *If $\langle S_1; S_2, s \rangle \Rightarrow^k s'$ then $\langle S_1, s \rangle \Rightarrow^{k_1} \hat{s}$ and $\langle S_2, \hat{s} \rangle \Rightarrow^{k_2} s'$, for some $\hat{s} \in \mathbf{State}$ and $k_1, k_2 > 0$ such that $k = k_1 + k_2$.*

Proof: By induction on k :

$k = 0$: OK

$k > 0$: Then $\langle S_1; S_2, s \rangle \Rightarrow \gamma \Rightarrow^{k-1} s'$. By the definition of the transitions, two possibilities only:

- $\gamma = \langle S_2, \hat{s} \rangle$, where $\langle S_1, s \rangle \Rightarrow \hat{s}$. OK
- $\gamma = \langle S'_1; S_2, s'' \rangle$, where $\langle S_1, s \rangle \Rightarrow \langle S'_1, s'' \rangle$. By the inductive hypothesis then, $\langle S'_1, s'' \rangle \Rightarrow^{k_1} \hat{s}$ and $\langle S_2, \hat{s} \rangle \Rightarrow^{k_2} s'$, for some $\hat{s} \in \mathbf{State}$ and $k_1, k_2 > 0$ such that $k - 1 = k_1 + k_2$. OK

Fact: *Further context does not influence computation:*

if $\langle S_1, s \rangle \Rightarrow^k \langle S'_1, s' \rangle$ then $\langle S_1; S_2, s \rangle \Rightarrow^k \langle S'_1; S_2, s' \rangle$;

if $\langle S_1, s \rangle \Rightarrow^k s'$ then $\langle S_1; S_2, s \rangle \Rightarrow^k \langle S_2, s' \rangle$.

Operational vs. natural semantics for TINY

“They are essentially the same”

Fact: *The two semantics are equivalent w.r.t. the final results described:*

$$\vdash \langle S, s \rangle \rightsquigarrow s' \text{ iff } \langle S, s \rangle \Rightarrow^* s'$$

for all statements $S \in \mathbf{Stmt}$ and states $s, s' \in \mathbf{State}$.

Proof:

“ $\Leftarrow$ ”: By induction on the length of the computation $\langle S, s \rangle \Rightarrow^* s'$.

“ $\Rightarrow$ ”: By induction on the structure of the derivation for $\langle S, s \rangle \rightsquigarrow s'$.

“ $\Leftarrow$ ”: By induction on the length of the computation $\langle S, s \rangle \Rightarrow^* s'$.

$\langle S, s \rangle \Rightarrow^k s'$: Take $k > 0$ and $\boxed{\langle S, s \rangle \Rightarrow \gamma \Rightarrow^{k-1} s'}$. By cases on the first step (few sample cases only):

- $\langle x := e, s \rangle \Rightarrow s[x \mapsto (\mathcal{E}[e] s)]$. Then $s' = s[x \mapsto (\mathcal{E}[e] s)]$;
 $\langle x := e, s \rangle \rightsquigarrow s[x \mapsto (\mathcal{E}[e] s)]$. OK
- $\langle S_1; S_2, s \rangle \Rightarrow \langle S'_1; S_2, s'' \rangle$, with $\langle S_1, s \rangle \Rightarrow \langle S'_1, s'' \rangle$.
Then $\langle S'_1; S_2, s'' \rangle \Rightarrow^{k-1} s'$, and so $\langle S'_1, s'' \rangle \Rightarrow^{k_1} \hat{s}''$ and $\langle S_2, \hat{s}'' \rangle \Rightarrow^{k_2} s'$, for $k_1, k_2 > 0$ with $k_1 + k_2 = k - 1$. Hence also $\langle S_1, s \rangle \Rightarrow^{k_1+1} \hat{s}''$.
Then $\langle S_1, s \rangle \rightsquigarrow \hat{s}''$ and $\langle S_2, \hat{s}'' \rangle \rightsquigarrow s'$, and so $\langle S_1; S_2, s \rangle \rightsquigarrow s'$. OK
- $\langle \text{if } b \text{ then } S_1 \text{ else } S_2, s \rangle \Rightarrow \langle S_1, s \rangle$, with $\mathcal{B}[b] s = \text{tt}$. Then
 $\langle S_1, s \rangle \Rightarrow^{k-1} s'$, so $\langle S_1, s \rangle \rightsquigarrow s'$ and $\langle \text{if } b \text{ then } S_1 \text{ else } S_2, s \rangle \rightsquigarrow s'$. OK
- $\langle \text{while } b \text{ do } S, s \rangle \Rightarrow \langle S; \text{while } b \text{ do } S, s \rangle$, with $\mathcal{B}[b] s = \text{tt}$. Then
 $\langle S; \text{while } b \text{ do } S, s \rangle \Rightarrow^{k-1} s'$, hence $\langle S, s \rangle \Rightarrow^{k_1} \hat{s}$ and
 $\langle \text{while } b \text{ do } S, \hat{s} \rangle \Rightarrow^{k_2} s'$, for $k_1, k_2 > 0$ with $k_1 + k_2 = k - 1$. Thus
 $\langle S, s \rangle \rightsquigarrow \hat{s}$, $\langle \text{while } b \text{ do } S, \hat{s} \rangle \rightsquigarrow s'$, and so $\langle \text{while } b \text{ do } S, s \rangle \rightsquigarrow s'$. OK

Induction on the structure of derivation trees

To prove **if** $\vdash \langle S, s \rangle \rightsquigarrow s'$ **then** $P(S, s, s')$ show:

clarify quantification

$$- P(x := e, s, s[x \mapsto (\mathcal{E}[e] s)])$$

$$\langle x := e, s \rangle \rightsquigarrow s[x \mapsto (\mathcal{E}[e] s)]$$

$$- P(\mathbf{skip}, s, s)$$

$$\langle \mathbf{skip}, s \rangle \rightsquigarrow s$$

$$\frac{\langle S_1, s \rangle \rightsquigarrow s' \quad \langle S_2, s' \rangle \rightsquigarrow s''}{\langle S_1; S_2, s \rangle \rightsquigarrow s''}$$

$$- P(S_1; S_2, s, s'') \text{ follows from } P(S_1, s, s') \text{ and } P(S_2, s', s'')$$

$$- P(\mathbf{if } b \mathbf{ then } S_1 \mathbf{ else } S_2, s, s') \text{ follows from } P(S_1, s, s') \text{ whenever } \mathcal{B}[b] s = \mathbf{tt}$$

$$\frac{\langle S_1, s \rangle \rightsquigarrow s' \quad \mathcal{B}[b] s = \mathbf{tt}}{\langle \mathbf{if } b \mathbf{ then } S_1 \mathbf{ else } S_2, s \rangle \rightsquigarrow s'}$$

$$\frac{\langle S_2, s \rangle \rightsquigarrow s' \quad \mathcal{B}[b] s = \mathbf{ff}}{\langle \mathbf{if } b \mathbf{ then } S_1 \mathbf{ else } S_2, s \rangle \rightsquigarrow s'}$$

$$- P(\mathbf{if } b \mathbf{ then } S_1 \mathbf{ else } S_2, s, s') \text{ follows from } P(S_2, s, s') \text{ whenever } \mathcal{B}[b] s = \mathbf{ff}$$

$$- P(\mathbf{while } b \mathbf{ do } S, s, s'') \text{ follows from } P(S, s, s') \text{ and } P(\mathbf{while } b \mathbf{ do } S, s', s'') \text{ whenever } \mathcal{B}[b] s = \mathbf{tt}$$

$$\frac{\mathcal{B}[b] s = \mathbf{tt} \quad \langle S, s \rangle \rightsquigarrow s' \quad \langle \mathbf{while } b \mathbf{ do } S, s' \rangle \rightsquigarrow s''}{\langle \mathbf{while } b \mathbf{ do } S, s \rangle \rightsquigarrow s''}$$

$$- P(\mathbf{while } b \mathbf{ do } S, s, s) \text{ whenever } \mathcal{B}[b] s = \mathbf{ff}$$

$$\frac{\mathcal{B}[b] s = \mathbf{ff}}{\langle \mathbf{while } b \mathbf{ do } S, s \rangle \rightsquigarrow s}$$

“ $\Rightarrow$ ”, i.e. **if** $\vdash \langle S, s \rangle \rightsquigarrow s'$ **then** $\langle S, s \rangle \Rightarrow^* s'$

By induction on the structure of the derivation for $\langle S, s \rangle \rightsquigarrow s'$.

- $\langle x := e, s \rangle \Rightarrow s[x \mapsto (\mathcal{E}[[e]] s)]$. OK
- $\langle \text{skip}, s \rangle \Rightarrow s$. OK
- Suppose $\langle S_1, s \rangle \rightsquigarrow s'$ and $\langle S_2, s' \rangle \rightsquigarrow s''$, so that $\langle S_1, s \rangle \Rightarrow^* s'$ and $\langle S_2, s' \rangle \Rightarrow^* s''$. Then $\langle S_1; S_2, s \rangle \Rightarrow^* \langle S_2, s' \rangle \Rightarrow^* s''$. OK
- Suppose $\mathcal{B}[[b]] s = \text{tt}$ and $\langle S_1, s \rangle \rightsquigarrow s'$, so that $\langle S_1, s \rangle \Rightarrow^* s'$. Then $\langle \text{if } b \text{ then } S_1 \text{ else } S_2, s \rangle \Rightarrow \langle S_1, s \rangle \Rightarrow^* s'$. OK
- Suppose $\mathcal{B}[[b]] s = \text{ff}$ and $\langle S_2, s \rangle \rightsquigarrow s'$, so that $\langle S_2, s \rangle \Rightarrow^* s'$. Then $\langle \text{if } b \text{ then } S_1 \text{ else } S_2, s \rangle \Rightarrow \langle S_2, s \rangle \Rightarrow^* s'$. OK
- Suppose $\mathcal{B}[[b]] s = \text{tt}$ and $\langle S, s \rangle \rightsquigarrow s'$ and $\langle \text{while } b \text{ do } S, s' \rangle \rightsquigarrow s''$, so that $\langle S, s \rangle \Rightarrow^* s'$ and $\langle \text{while } b \text{ do } S, s' \rangle \Rightarrow^* s''$. Then $\langle \text{while } b \text{ do } S, s \rangle \Rightarrow \langle S; \text{while } b \text{ do } S, s \rangle \Rightarrow^* \langle \text{while } b \text{ do } S, s' \rangle \Rightarrow^* s''$. OK
- If $\mathcal{B}[[b]] s = \text{ff}$ then $\langle \text{while } b \text{ do } S, s \rangle \Rightarrow s$. OK

Adequacy of denotational semantics

Fact: For each statement $S \in \mathbf{Stmt}$ and states $s, s' \in \mathbf{State}$,

$$\langle S, s \rangle \Rightarrow^* s' \text{ iff } \mathcal{S}[[S]] s = s'$$

Proof:

“ $\Rightarrow$ ”: By structural induction on S , then by induction on the length of the computation $\langle S, s \rangle \Rightarrow^* s'$.

“ $\Leftarrow$ ”: By structural induction on S .

BTW: In the proof of either implication, the only interesting case is that of loops — we omit the other cases.

$$\langle \text{while } b \text{ do } S, s \rangle \Rightarrow^* s' \implies \text{for some } n \geq 0, \Phi^n(\emptyset_{\text{State} \rightarrow \text{State}}) s = s'$$

$$\text{where } \Phi(F) = \text{cond}(\mathcal{B}[[b]], \mathcal{S}[[S]]; F, \text{id}_{\text{State}})$$

Relying on the inductive hypothesis $\langle S, s \rangle \Rightarrow^* \hat{s} \implies \mathcal{S}[[S]] s = \hat{s}$,
 by induction on the length of the computation $\langle \text{while } b \text{ do } S, s \rangle \Rightarrow^k s'$.

$k > 0$: Then $\langle \text{while } b \text{ do } S, s \rangle \Rightarrow \gamma \Rightarrow^{k-1} s'$. By cases on this first step:

- $\mathcal{B}[[b]] s = \text{ff}$ and $\gamma = s$. Then $s' = s$, and $\Phi(\emptyset_{\text{State} \rightarrow \text{State}}) s = s$. OK
- $\mathcal{B}[[b]] s = \text{tt}$ and $\gamma = \langle S; \text{while } b \text{ do } S, s \rangle \Rightarrow^{k-1} s'$. Then $\langle S, s \rangle \Rightarrow^{k_1} \hat{s}$ and $\langle \text{while } b \text{ do } S, \hat{s} \rangle \Rightarrow^{k_2} s'$, for some $\hat{s} \in \text{State}$ and $k_1, k_2 > 0$ with $k_1 + k_2 = k - 1$. Hence, $\mathcal{S}[[S]] s = \hat{s}$ and $\Phi^n(\emptyset_{\text{State} \rightarrow \text{State}}) \hat{s} = s'$ for some $n \geq 0$. Thus, $\Phi^{n+1}(\emptyset_{\text{State} \rightarrow \text{State}}) s = s'$. OK

$$\langle \text{while } b \text{ do } S, s \rangle \Rightarrow^* s' \iff \text{for some } n \geq 0, \Phi^n(\emptyset_{\text{State} \rightarrow \text{State}}) s = s' \\ \text{where } \Phi(F) = \text{cond}(\mathcal{B}[b], \mathcal{S}[S]; F, \text{id}_{\text{State}})$$

Relying on the inductive hypothesis $\langle S, s \rangle \Rightarrow^* \hat{s} \iff \mathcal{S}[S] s = \hat{s}$,
by induction on $n \geq 0$, assuming $\Phi^n(\emptyset_{\text{State} \rightarrow \text{State}}) s = s'$.

$n > 0$: Then $\Phi^n(\emptyset_{\text{State} \rightarrow \text{State}}) s = \text{cond}(\mathcal{B}[b], \mathcal{S}[S]; \Phi^{n-1}(\emptyset_{\text{State} \rightarrow \text{State}}), \text{id}_{\text{State}}) s$.

- $\mathcal{B}[b] s = \text{ff}$: then $\Phi^n(\emptyset_{\text{State} \rightarrow \text{State}}) s = s$, so $s' = s$, and also $\langle \text{while } b \text{ do } S, s \rangle \Rightarrow s$. OK
- $\mathcal{B}[b] s = \text{tt}$: then $\Phi^n(\emptyset_{\text{State} \rightarrow \text{State}}) s = \Phi^{n-1}(\emptyset_{\text{State} \rightarrow \text{State}})(\hat{s}) = s'$, where $\hat{s} = \mathcal{S}[S] s$. Hence, $\langle \text{while } b \text{ do } S, \hat{s} \rangle \Rightarrow^* s'$, and since $\langle S, s \rangle \Rightarrow^* \hat{s}$, we get $\langle \text{while } b \text{ do } S, s \rangle \Rightarrow \langle S; \text{while } b \text{ do } S, s \rangle \Rightarrow^* \langle \text{while } b \text{ do } S, \hat{s} \rangle \Rightarrow^* s'$. OK

Soundness of Hoare's proof calculus

if $\mathcal{TH}(\mathbf{Int}) \vdash \{\varphi\} S \{\psi\}$ then $\models \{\varphi\} S \{\psi\}$, i.e. $\{\varphi\} \llbracket S \rrbracket \subseteq \{\psi\}$

By induction on the structure of the proof in Hoare's logic:

assignment rule: Easy, but we need a lemma (proof by structural induction on the formulae): $\mathcal{F} \llbracket \varphi[x \mapsto e] \rrbracket s = \mathcal{F} \llbracket \varphi \rrbracket s[x \mapsto \mathcal{E} \llbracket e \rrbracket s]$.

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

Then, for $s \in \mathbf{State}$, if $s \in \{\varphi[x \mapsto e]\}$ then $\mathcal{S} \llbracket x := e \rrbracket s = s[x \mapsto \mathcal{E} \llbracket e \rrbracket s] \in \{\varphi\}$.

skip rule: Trivial.

composition rule: Assume $\{\varphi\} \llbracket S_1 \rrbracket \subseteq \{\theta\}$ and $\{\theta\} \llbracket S_2 \rrbracket \subseteq \{\psi\}$. Then

$$\{\varphi\} \llbracket S_1; S_2 \rrbracket = (\{\varphi\} \llbracket S_1 \rrbracket) \llbracket S_2 \rrbracket \subseteq \{\theta\} \llbracket S_2 \rrbracket \subseteq \{\psi\}.$$

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

if-then-else rule: Easy.

consequence rule: Again the same, given the obvious observation that $\{\varphi_1\} \subseteq \{\varphi_2\}$

iff $\varphi_1 \Rightarrow \varphi_2 \in \mathcal{TH}(\mathbf{Int})$.

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

Soundness of the loop rule

loop rule: We need to show that the least fixed point of the operator

$$\Phi(F) = \text{cond}(\mathcal{B}[\![b]\!], \mathcal{S}[\![S]\!]; F, \text{id}_{\mathbf{State}})$$

satisfies

$$\text{fix}(\Phi)(\{\varphi\}) \subseteq \{\varphi \wedge \neg b\}$$

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

Proceed by fixed point induction (*this is an admissible property!*).

Suppose that $F(\{\varphi\}) \subseteq \{\varphi \wedge \neg b\}$ for some $F: \mathbf{State} \rightarrow \mathbf{State}$, and consider $s \in \{\varphi\}$ with $s' = \Phi(F)(s) \in \mathbf{State}$. Two cases are possible:

- If $\mathcal{B}[\![b]\!] s = \mathbf{ff}$ then $s' = s \in \{\varphi \wedge \neg b\}$.
- If $\mathcal{B}[\![b]\!] s = \mathbf{tt}$ then $s' = F(\mathcal{S}[\![S]\!] s)$. We get $s' \in \{\varphi \wedge \neg b\}$ by the assumption on F , since $\{\varphi \wedge b\} \llbracket S \rrbracket \subseteq \{\varphi\}$ by the inductive hypothesis, which implies $\mathcal{S}[\![S]\!] s \in \{\varphi\}$.

So, $\Phi(F)(\{\varphi\}) \subseteq \{\varphi \wedge \neg b\}$, and the proof is completed.

Further properties

- completeness of Hoare's proof calculus
- soundness and completeness of proof calculus for total correctness

to be discussed later...