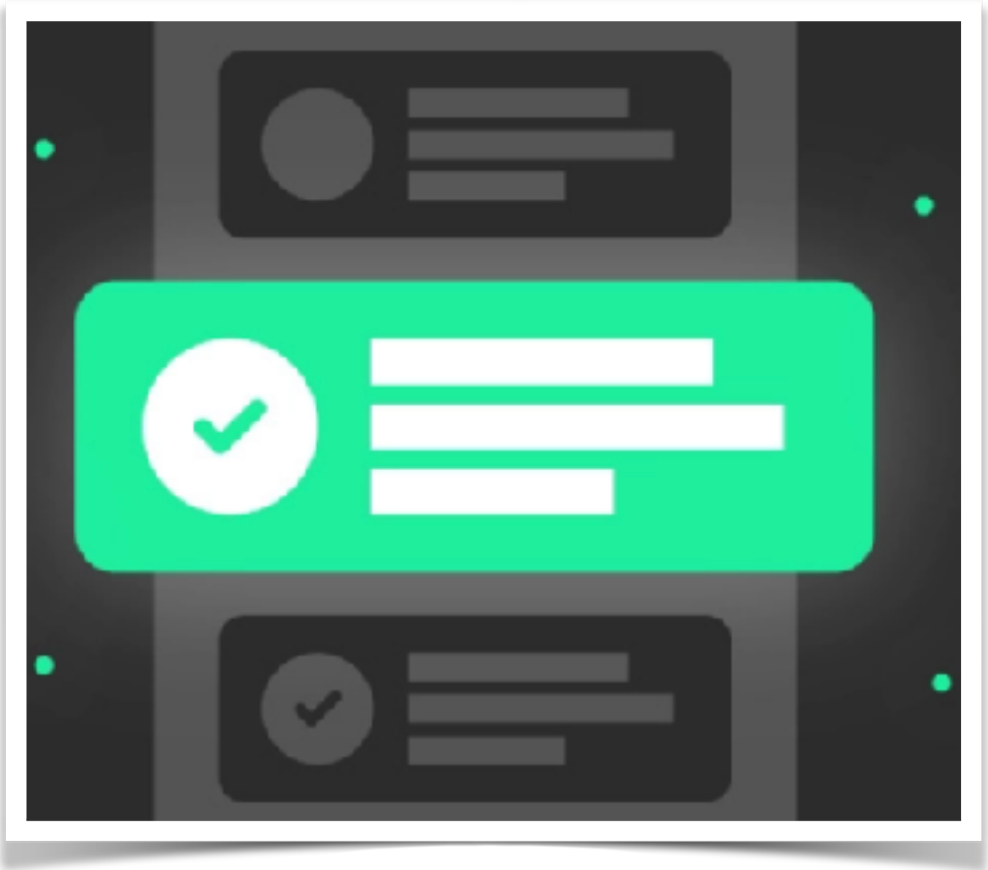

http://didawiki.di.unipi.it/doku.php/magistraleinformatica/psc/start

PSC 2022/23 (375AA, 9CFU)

Principles for Software Composition

Roberto Bruni
http://www.di.unipi.it/~bruni/

10 - Consistency and congruence

Operational equivalence

Operational equivalence

$$a_1 \sim_{\mathrm{op}} a_2 \quad \text{iff} \quad \forall \sigma, n.\ (\ \langle a_1, \sigma \rangle \to n \Leftrightarrow \langle a_2, \sigma \rangle \to n\ )$$

$$b_1 \sim_{\mathrm{op}} b_2 \quad \text{iff} \quad \forall \sigma, v.\ (\ \langle b_1, \sigma \rangle \to v \Leftrightarrow \langle b_2, \sigma \rangle \to v\ )$$

$$c_1 \sim_{\mathrm{op}} c_2 \quad \text{iff} \quad \forall \sigma, \sigma'.\ (\ \langle c_1, \sigma \rangle \to \sigma' \Leftrightarrow \langle c_2, \sigma \rangle \to \sigma'\ )$$

termination and determinacy does not matter:
operational equivalence is always well-defined

Congruence

$$a_1 \sim_{\mathrm{op}} a_2 \quad \text{iff} \quad \forall \sigma, n.\ (\ \langle a_1, \sigma \rangle \rightarrow n \Leftrightarrow \langle a_2, \sigma \rangle \rightarrow n\ )$$

take any context $\mathbb{A}[\cdot]$ e.g. $2 \times ([\cdot] + 5)$

is it the case that $a_1 \sim_{\mathrm{op}} a_2 \Rightarrow \mathbb{A}[a_1] \sim_{\mathrm{op}} \mathbb{A}[a_2]$?

that is: can we replace a subexpressions with
any equivalent one without changing the outcome?

Contexts

what are the possible contexts for arithmetic expressions?

$$[\cdot] + 5$$

$$2 \times ([\cdot] + 5)$$

$$2 \times ([\cdot] + 5) \leq 50$$

$$(2 \times ([\cdot] + 5) \leq 50) \wedge x = y$$

$$x := 2 \times ([\cdot] + 5)$$

$$\mathbf{while}\ x \leq 100\ \mathbf{do}\ x := 2 \times ([\cdot] + 5)$$

Contexts

what are the possible contexts for arithmetic expressions?

$$
\begin{array}{rcl}
\mathbb{A}[\cdot] & ::= & [\cdot] \\
& | & \mathbb{A}[\cdot] \ \mathsf{op} \ a \\
& | & a \ \mathsf{op} \ \mathbb{A}[\cdot]
\end{array}
$$

$$
\begin{array}{rcl}
\mathbb{B}[\cdot] & ::= & \mathbb{A}[\cdot] \ \mathsf{cmp} \ a \\
& | & a \ \mathsf{cmp} \ \mathbb{A}[\cdot] \\
& | & \neg \mathbb{B}[\cdot] \\
& | & \mathbb{B}[\cdot] \ \mathsf{bop} \ b \\
& | & b \ \mathsf{bop} \ \mathbb{B}[\cdot]
\end{array}
$$

$$
\begin{array}{rcl}
\mathbb{C}[\cdot] & ::= & x := \mathbb{A}[\cdot] \\
& | & \mathbb{C}[\cdot]; c \\
& | & c; \mathbb{C}[\cdot] \\
& | & \textbf{if } \mathbb{B}[\cdot] \textbf{ then } c \textbf{ else } c \\
& | & \textbf{if } b \textbf{ then } \mathbb{C}[\cdot] \textbf{ else } c \\
& | & \textbf{if } b \textbf{ then } c \textbf{ else } \mathbb{C}[\cdot] \\
& | & \textbf{while } \mathbb{B}[\cdot] \textbf{ do } c \\
& | & \textbf{while } b \textbf{ do } \mathbb{C}[\cdot]
\end{array}
$$

Proof obligations

many proof obligations to deal with:

$$\forall a, a_1, a_2.\ (\ a_1 \sim_{\text{op}} a_2 \Rightarrow a_1 \ \mathsf{op}\ a \sim_{\text{op}} a_2 \ \mathsf{op}\ a\ )$$

$$\forall a, a_1, a_2.\ (\ a_1 \sim_{\text{op}} a_2 \Rightarrow a \ \mathsf{op}\ a_1 \sim_{\text{op}} a \ \mathsf{op}\ a_2\ )$$

$$\forall a, a_1, a_2.\ (\ a_1 \sim_{\text{op}} a_2 \Rightarrow a \ \mathsf{cmp}\ a_1 \sim_{\text{op}} a \ \mathsf{cmp}\ a_2\ )$$

$$\forall a, a_1, a_2.\ (\ a_1 \sim_{\text{op}} a_2 \Rightarrow a_1 \ \mathsf{cmp}\ a \sim_{\text{op}} a_2 \ \mathsf{cmp}\ a\ )$$

$$\forall x, a_1, a_2.\ (\ a_1 \sim_{\text{op}} a_2 \Rightarrow x := a_1 \sim_{\text{op}} x := a_2\ )$$

similarly for boolean expressions and commands

Denotational equivalence

Denotational equivalence

$$a_1 \sim_{\mathrm{den}} a_2 \quad \text{iff} \quad \mathcal{A}[\![a_1]\!] = \mathcal{A}[\![a_2]\!]$$

$$b_1 \sim_{\mathrm{den}} b_2 \quad \text{iff} \quad \mathcal{B}[\![b_1]\!] = \mathcal{B}[\![b_2]\!]$$

$$c_1 \sim_{\mathrm{den}} c_2 \quad \text{iff} \quad \mathcal{C}[\![c_1]\!] = \mathcal{C}[\![c_2]\!]$$

(two functions are the same
if they coincide on all arguments)

Compositionality principle

$$a_1 \sim_{\text{den}} a_2 \quad \text{iff} \quad \mathcal{A}[\![a_1]\!] = \mathcal{A}[\![a_2]\!]$$

take any context $\mathbb{A}[\cdot]$

is it the case that $a_1 \sim_{\text{den}} a_2 \Rightarrow \mathbb{A}[a_1] \sim_{\text{den}} \mathbb{A}[a_2]$?

YES! it is guaranteed by the compositionally principle of denotational semantics:

the meaning of a compound expression is solely determined by the meaning of its constituents

Consistency

if we guarantee the consistency between
the operational semantics and
the denotational semantics
then the congruence property is guaranteed
for the operational semantics too

$$\forall a_1, a_2.\ (\ a_1 \sim_{\mathrm{op}} a_2 \overset{?}{\Leftrightarrow} a_1 \sim_{\mathrm{den}} a_2\ )$$

$$\forall b_1, b_2.\ (\ b_1 \sim_{\mathrm{op}} b_2 \overset{?}{\Leftrightarrow} b_1 \sim_{\mathrm{den}} b_2\ )$$

$$\forall c_1, c_2.\ (\ c_1 \sim_{\mathrm{op}} c_2 \overset{?}{\Leftrightarrow} c_1 \sim_{\mathrm{den}} c_2\ )$$

Consistency: expressions

$$\forall\, a \in Aexp\ \forall \sigma \in \Sigma.\ \langle a, \sigma \rangle \to \mathscr{A}\,[\![a]\!]\,\sigma$$

$$P(a) \stackrel{\text{def}}{=} \forall \sigma \in \Sigma.\ \langle a, \sigma \rangle \to \mathscr{A}\,[\![a]\!]\,\sigma$$

by structural induction

$$\forall\, b \in Bexp\ \forall \sigma \in \Sigma.\ \langle b, \sigma \rangle \to \mathscr{B}\,[\![b]\!]\,\sigma$$

$$P(b) \stackrel{\text{def}}{=} \forall \sigma \in \Sigma.\ \langle b, \sigma \rangle \to \mathscr{B}\,[\![b]\!]\,\sigma$$

by structural induction

Consistency: commands

$\forall c \in Com.\ \forall \sigma, \sigma' \in \Sigma.$ $\quad \langle c, \sigma \rangle \to \sigma' \quad \Leftrightarrow \quad \mathscr{C}\,[\![c]\!]\,\sigma = \sigma'$

can we write it as

$\forall c \in Com.\ \forall \sigma \in \Sigma. \quad \langle c, \sigma \rangle \to \mathscr{C}\,[\![c]\!]\,\sigma$?

no, because there is no such formula as

$$\langle c, \sigma \rangle \to \perp$$

Consistency: commands

$$\forall c \in Com.\ \forall \sigma, \sigma' \in \Sigma. \quad \langle c, \sigma \rangle \rightarrow \sigma' \quad \Leftrightarrow \quad \mathscr{C}\,[\![c]\!]\,\sigma = \sigma'$$

$$\forall c \in Com.\ \forall \sigma, \sigma' \in \Sigma.$$

Correctness

$$P(\langle c, \sigma \rangle \rightarrow \sigma') \stackrel{\text{def}}{=} \mathscr{C}\,[\![c]\!]\,\sigma = \sigma'$$

by rule induction

$$\forall c \in Com.$$

Completeness

$$P(c) \stackrel{\text{def}}{=} \forall \sigma, \sigma' \in \Sigma. \quad \mathscr{C}\,[\![c]\!]\,\sigma = \sigma' \quad \Rightarrow \quad \langle c, \sigma \rangle \rightarrow \sigma'$$

by structural induction

Correctness

$$\forall c \in Com,\ \forall \sigma, \sigma' \in \Sigma$$

$$P(\langle c, \sigma \rangle \to \sigma') \stackrel{\text{def}}{=} \mathscr{C}[\![c]\!]\sigma = \sigma'$$

by rule induction

$$\frac{}{\langle \mathbf{skip}, \sigma \rangle \to \sigma}$$

We want to prove

$$P(\langle \mathbf{skip}, \sigma \rangle \to \sigma) \stackrel{\text{def}}{=} \mathscr{C}\,[\![\mathbf{skip}]\!]\,\sigma = \sigma$$

Obviously the proposition is true by the definition of the denotational semantics.

$$\frac{\langle a, \sigma \rangle \to m}{\langle x := a, \sigma \rangle \to \sigma\left[{}^{m}/_{x}\right]}$$

We assume $\langle a, \sigma \rangle \to m$ and hence $\mathscr{A}\,[\![a]\!]\,\sigma = m$ by the equivalence of the operational and denotational semantics of arithmetic expressions.
We want to prove

$$P(\langle x := a, \sigma \rangle \to \sigma\left[{}^{m}/_{x}\right]) \stackrel{\text{def}}{=} \mathscr{C}\,[\![x := a]\!]\,\sigma = \sigma\left[{}^{m}/_{x}\right]$$

By the definition of the denotational semantics

$$\mathscr{C}\,[\![x := a]\!]\,\sigma = \sigma\left[{}^{\mathscr{A}\,[\![a]\!]\,\sigma}/_{x}\right] = \sigma\left[{}^{m}/_{x}\right]$$

$$\frac{\langle c_0, \sigma \rangle \to \sigma'' \quad \langle c_1, \sigma'' \rangle \to \sigma'}{\langle c_0; c_1, \sigma \rangle \to \sigma'}$$

We assume

$$P(\langle c_0, \sigma \rangle \to \sigma'') \stackrel{\text{def}}{=} \mathscr{C}\,[\![c_0]\!]\,\sigma = \sigma''$$
$$P(\langle c_1, \sigma'' \rangle \to \sigma') \stackrel{\text{def}}{=} \mathscr{C}\,[\![c_1]\!]\,\sigma'' = \sigma'$$

We want to prove

$$P(\langle c_0; c_1, \sigma \rangle \to \sigma') \stackrel{\text{def}}{=} \mathscr{C}\,[\![c_0; c_1]\!]\,\sigma = \sigma'$$

By the denotational semantics definition and the inductive hypotheses

$$\mathscr{C}\,[\![c_0; c_1]\!]\,\sigma = \mathscr{C}\,[\![c_1]\!]^*\,(\mathscr{C}\,[\![c_0]\!]\,\sigma) = \mathscr{C}\,[\![c_1]\!]^*\,\sigma'' = \mathscr{C}\,[\![c_1]\!]\,\sigma'' = \sigma'$$

Note that the lifting operator can be removed because $\sigma'' \neq \bot$ by the inductive hypothesis.

$$\frac{\langle b, \sigma\rangle \to \textbf{true} \quad \langle c_0, \sigma\rangle \to \sigma'}{\langle \textbf{if}\ b\ \textbf{then}\ c_0\ \textbf{else}\ c_1, \sigma\rangle \to \sigma'}$$

We assume

- $\langle b, \sigma\rangle \to \textbf{true}$ and therefore $\mathscr{B}[\![b]\!]\,\sigma = \textbf{true}$ by the correspondence between the operational and denotational semantics for boolean expressions;
- $P(\langle c_0, \sigma\rangle \to \sigma') \stackrel{\text{def}}{=} \mathscr{C}[\![c_0]\!]\,\sigma = \sigma'$

We want to prove

$$P(\langle \textbf{if}\ b\ \textbf{then}\ c_0\ \textbf{else}\ c_1, \sigma\rangle \to \sigma') \stackrel{\text{def}}{=} \mathscr{C}[\![\textbf{if}\ b\ \textbf{then}\ c_0\ \textbf{else}\ c_1]\!]\,\sigma = \sigma'$$

In fact, we have

$$\begin{aligned}
\mathscr{C}[\![\textbf{if}\ b\ \textbf{then}\ c_0\ \textbf{else}\ c_1]\!]\,\sigma &= \mathscr{B}[\![b]\!]\,\sigma \to \mathscr{C}[\![c_0]\!]\,\sigma, \mathscr{C}[\![c_1]\!]\,\sigma \\
&= \textbf{true} \to \sigma', \mathscr{C}[\![c_1]\!]\,\sigma \\
&= \sigma'
\end{aligned}$$

$$\frac{\langle b, \sigma \rangle \to \mathbf{false}}{\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma \rangle \to \sigma}$$

We assume $\langle b, \sigma \rangle \to \mathbf{false}$ and therefore $\mathscr{B}\,[\![b]\!]\,\sigma = \mathbf{false}$.
We want to prove

$$P(\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma \rangle \to \sigma) \stackrel{\text{def}}{=} \mathscr{C}\,[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]\,\sigma = \sigma$$

By the fixpoint property of the denotational semantics

$$\begin{aligned}
\mathscr{C}\,[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]\,\sigma &= \mathscr{B}\,[\![b]\!]\,\sigma \to \mathscr{C}\,[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]^{*}\,(\mathscr{C}\,[\![c]\!]\,\sigma), \sigma \\
&= \mathbf{false} \to \mathscr{C}\,[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]^{*}\,(\mathscr{C}\,[\![c]\!]\,\sigma), \sigma \\
&= \sigma
\end{aligned}$$

$$\frac{\langle b,\sigma\rangle \to \mathbf{true} \quad \langle c,\sigma\rangle \to \sigma'' \quad \langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma''\rangle \to \sigma'}{\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma\rangle \to \sigma'}$$

We assume

- $\langle b,\sigma\rangle \to \mathbf{true}$ and therefore $\mathscr{B}[\![b]\!]\,\sigma = \mathbf{true}$
- $P(\langle c,\sigma\rangle \to \sigma'') \stackrel{\text{def}}{=} \mathscr{C}[\![c]\!]\,\sigma = \sigma''$
- $P(\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma''\rangle \to \sigma') \stackrel{\text{def}}{=} \mathscr{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]\,\sigma'' = \sigma'$

We want to prove

$$P(\langle \mathbf{while}\ b\ \mathbf{do}\ c, \sigma\rangle \to \sigma') \stackrel{\text{def}}{=} \mathscr{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]\,\sigma = \sigma'$$

By the definition of the denotational semantics and the inductive hypotheses

$$\begin{aligned}
\mathscr{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]\,\sigma &= \mathscr{B}[\![b]\!]\,\sigma \to \mathscr{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]^*\,(\mathscr{C}[\![c]\!]\,\sigma), \sigma \\
&= \mathbf{true} \to \mathscr{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]^*\,\sigma'', \sigma \\
&= \mathscr{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]^*\,\sigma'' \\
&= \mathscr{C}[\![\mathbf{while}\ b\ \mathbf{do}\ c]\!]\,\sigma'' \\
&= \sigma'
\end{aligned}$$

Note that the lifting operator can be removed since $\sigma'' \neq \bot$.

Completeness

$$\forall c \in Com$$

$$P(c) \stackrel{\text{def}}{=} \forall \boldsymbol{\sigma}, \boldsymbol{\sigma}' \in \Sigma. \quad \mathscr{C}\,[\![c]\!]\,\boldsymbol{\sigma} = \boldsymbol{\sigma}' \quad \Rightarrow \quad \langle c, \boldsymbol{\sigma} \rangle \to \boldsymbol{\sigma}'$$

by structural induction

We prove $P(\mathbf{skip}) \stackrel{\text{def}}{=} \forall \sigma, \sigma'. \mathscr{C}\,[\![\mathbf{skip}]\!]\,\sigma = \sigma' \Rightarrow \langle \mathbf{skip}, \sigma \rangle \to \sigma'$

Assume $\mathscr{C}\,[\![\mathbf{skip}]\!]\,\sigma = \sigma'$

Then $\sigma' = \sigma$

By rule (skip) $\langle \mathbf{skip}, \sigma \rangle \to \sigma = \sigma'$

We prove $P(x := a) \stackrel{\text{def}}{=} \forall \sigma, \sigma'.\, \mathscr{C}[\![x := a]\!]\,\sigma = \sigma' \Rightarrow \langle x := a, \sigma\rangle \to \sigma'$

Assume $\mathscr{C}[\![x := a]\!]\,\sigma = \sigma'$

Then $\sigma' = \sigma[{}^{\mathscr{A}[\![a]\!]\sigma}/_{x}]$

By consistency for expressions $\langle a, \sigma\rangle \to \mathscr{A}[\![a]\!]\,\sigma$

By rule (asgn) $\langle x := a, \sigma\rangle \to \sigma[{}^{\mathscr{A}[\![a]\!]\sigma}/_{x}] = \sigma'$

Assume $P(c_0) \stackrel{\text{def}}{=} \forall \sigma, \sigma''.\ \mathscr{C}\llbracket c_0 \rrbracket \sigma = \sigma'' \Rightarrow \langle c_0, \sigma \rangle \to \sigma''$

$P(c_1) \stackrel{\text{def}}{=} \forall \sigma'', \sigma'.\ \mathscr{C}\llbracket c_1 \rrbracket \sigma'' = \sigma' \Rightarrow \langle c_1, \sigma'' \rangle \to \sigma'$

We want to prove $P(c_0;c_1) \stackrel{\text{def}}{=} \forall \sigma, \sigma'.\ \mathscr{C}\llbracket c_0;c_1 \rrbracket \sigma = \sigma' \Rightarrow \langle c_0;c_1, \sigma \rangle \to \sigma'$

Assume $\mathscr{C}\llbracket c_0;c_1 \rrbracket \sigma = \sigma'$

we have $\mathscr{C}\llbracket c_0;c_1 \rrbracket \sigma = \mathscr{C}\llbracket c_1 \rrbracket^* (\mathscr{C}\llbracket c_0 \rrbracket \sigma) = \sigma' \neq \perp$

thus $\mathscr{C}\llbracket c_0 \rrbracket \sigma = \sigma''$ for some $\sigma'' \neq \perp$

and $\mathscr{C}\llbracket c_1 \rrbracket \sigma'' = \sigma'$

by inductive hypotheses $\langle c_0, \sigma \rangle \to \sigma''$ $\langle c_1, \sigma'' \rangle \to \sigma'$

By rule (seq) $\langle c_0;c_1, \sigma \rangle \to \sigma'$

Assume $P(c_0) \stackrel{\text{def}}{=} \forall \sigma, \sigma'. \, \mathscr{C}\llbracket c_0 \rrbracket \sigma = \sigma' \Rightarrow \langle c_0, \sigma \rangle \rightarrow \sigma'$

$P(c_1) \stackrel{\text{def}}{=} \forall \sigma, \sigma'. \, \mathscr{C}\llbracket c_1 \rrbracket \sigma = \sigma' \Rightarrow \langle c_1, \sigma \rangle \rightarrow \sigma'$

We prove $P(\textbf{if } b \textbf{ then } c_0 \textbf{ else } c_1) \stackrel{\text{def}}{=} \forall \sigma, \sigma'. \, \mathscr{C}\llbracket \textbf{if } b \textbf{ then } c_0 \textbf{ else } c_1 \rrbracket \sigma = \sigma'$
$\Rightarrow \langle \textbf{if } b \textbf{ then } c_0 \textbf{ else } c_1, \sigma \rangle \rightarrow \sigma'$

Assume $\mathscr{C}\llbracket \textbf{if } b \textbf{ then } c_0 \textbf{ else } c_1 \rrbracket \sigma = \sigma'$

we have $\mathscr{C}\llbracket \textbf{if } b \textbf{ then } c_0 \textbf{ else } c_1 \rrbracket \sigma = \mathscr{B}\llbracket b \rrbracket \sigma \rightarrow \mathscr{C}\llbracket c_0 \rrbracket \sigma, \mathscr{C}\llbracket c_1 \rrbracket \sigma = \sigma'$

either $\mathscr{B}\llbracket b \rrbracket \sigma = \textbf{false}$ or $\mathscr{B}\llbracket b \rrbracket \sigma = \textbf{true}$

if $\mathscr{B}\llbracket b \rrbracket \sigma = \textbf{false}$ $\mathscr{C}\llbracket \textbf{if } b \textbf{ then } c_0 \textbf{ else } c_1 \rrbracket \sigma = \mathscr{C}\llbracket c_1 \rrbracket \sigma = \sigma'$

$\langle b, \sigma \rangle \rightarrow \textbf{false}$ by inductive hypotheses $\langle c_1, \sigma \rangle \rightarrow \sigma'$

By rule (ifff) $\langle \textbf{if } b \textbf{ then } c_0 \textbf{ else } c_1, \sigma \rangle \rightarrow \sigma'$

if $\mathscr{B}\llbracket b \rrbracket \sigma = \textbf{true}$ $\mathscr{C}\llbracket \textbf{if } b \textbf{ then } c_0 \textbf{ else } c_1 \rrbracket \sigma = \mathscr{C}\llbracket c_0 \rrbracket \sigma = \sigma'$

$\langle b, \sigma \rangle \rightarrow \textbf{true}$ by inductive hypotheses $\langle c_0, \sigma \rangle \rightarrow \sigma'$

By rule (iftt) $\langle \textbf{if } b \textbf{ then } c_0 \textbf{ else } c_1, \sigma \rangle \rightarrow \sigma'$

Assume $P(c) \stackrel{\text{def}}{=} \forall \sigma, \sigma''.\ \mathscr{C}\,[\![c]\!]\,\sigma = \sigma'' \Rightarrow \langle c, \sigma \rangle \to \sigma''$

We prove $P(\textbf{while } b \textbf{ do } c) \stackrel{\text{def}}{=} \forall \sigma, \sigma'.\ \mathscr{C}\,[\![\textbf{while } b \textbf{ do } c]\!]\,\sigma = \sigma' \Rightarrow \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma'$

we have $\mathscr{C}\,[\![\textbf{while } b \textbf{ do } c]\!]\,\sigma = \text{fix } \Gamma_{b,c}\,\sigma = \left(\bigsqcup_{n\in\mathbb{N}} \Gamma_{b,c}^{n} \bot \right) \sigma$

$$\mathscr{C}\,[\![\textbf{while } b \textbf{ do } c]\!]\,\sigma = \sigma' \Rightarrow \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma'$$

iff $\left(\bigsqcup_{n\in\mathbb{N}} \Gamma_{b,c}^{n} \bot \right) \sigma = \sigma' \Rightarrow \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma'$

iff $\left(\exists n \in \mathbb{N}.\ (\Gamma_{b,c}^{n} \bot) \sigma = \sigma' \right) \Rightarrow \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma'$

iff $\forall n \in \mathbb{N}.\ \left(\Gamma_{b,c}^{n} \bot \sigma = \sigma' \Rightarrow \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma' \right)$

let $A(n) \stackrel{\text{def}}{=} \forall \sigma, \sigma'.\ \Gamma_{b,c}^{n} \bot \sigma = \sigma' \Rightarrow \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma'$

we prove $\forall n \in \mathbb{N}.\ A(n)$ by mathematical induction

Assume $P(c) \stackrel{\text{def}}{=} \forall \sigma, \sigma''. \mathscr{C}\,[\![c]\!]\,\sigma = \sigma'' \Rightarrow \langle c, \sigma \rangle \to \sigma''$

we prove $\forall n \in \mathbb{N}. A(n) \stackrel{\text{def}}{=} \forall \sigma, \sigma'. \Gamma^{n}_{b,c} \bot \sigma = \sigma' \Rightarrow \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma'$

$A(0) \stackrel{\text{def}}{=} \forall \sigma, \sigma'. \Gamma^{0}_{b,c} \bot \sigma = \sigma' \Rightarrow \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma'$

$\Gamma^{0}_{b,c} \bot \sigma = \bot \sigma = \bot$

the premise $\Gamma^{0}_{b,c} \bot \sigma = \sigma'$ is false $\quad \sigma' \neq \bot$

A(0) is true

Assume $P(c) \stackrel{\text{def}}{=} \forall \sigma, \sigma''.\ \mathscr{C}[\![c]\!]\sigma = \sigma'' \Rightarrow \langle c, \sigma \rangle \to \sigma''$

we prove $\forall n \in \mathbb{N}.\ A(n) \stackrel{\text{def}}{=} \forall \sigma, \sigma'.\ \Gamma^n_{b,c} \bot \sigma = \sigma' \Rightarrow \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma'$

assume $A(n) \stackrel{\text{def}}{=} \forall \sigma, \sigma'.\ \Gamma^n_{b,c} \bot \sigma = \sigma' \Rightarrow \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma'$

we prove $A(n+1) \stackrel{\text{def}}{=} \forall \sigma, \sigma'.\ \Gamma^{n+1}_{b,c} \bot \sigma = \sigma' \Rightarrow \langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma'$

assume $\Gamma^{n+1}_{b,c} \bot \sigma = \Gamma_{b,c}\left(\Gamma^n_{b,c} \bot\right) \sigma = \sigma' \neq \bot$

by def $\mathscr{B}[\![b]\!]\sigma \to \left(\Gamma^n_{b,c} \bot\right)^* (\mathscr{C}[\![c]\!]\sigma), \sigma = \sigma'$

if $\mathscr{B}[\![b]\!]\sigma = \textbf{false}$ $\quad \langle b, \sigma \rangle \to \textbf{false}$ $\quad \sigma = \sigma'$

by rule (whff)

$\langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma$
$= \sigma'$

if $\mathscr{B}[\![b]\!]\sigma = \textbf{true}$ $\quad \langle b, \sigma \rangle \to \textbf{true}$ $\quad \left(\Gamma^n_{b,c} \bot\right)^* (\mathscr{C}[\![c]\!]\sigma) = \sigma' \neq \bot$

thus $\mathscr{C}[\![c]\!]\sigma = \sigma''$ for some $\sigma'' \neq \bot$

$\langle c, \sigma \rangle \to \sigma''$

$\left(\Gamma^n_{b,c} \bot\right) \sigma'' = \sigma'$

$\langle \textbf{while } b \textbf{ do } c, \sigma'' \rangle \to \sigma'$

By rule (whtt)

$\langle \textbf{while } b \textbf{ do } c, \sigma \rangle \to \sigma'$

Final remarks

Commands

Big-step operational semantics

Termination ✗

Determinacy ✓

Operational equivalence

Denotational semantics

(partial functions)

Denotational equivalence is a congruence

Consistency
(correctness + completeness)

Operational equivalence = Denotational equivalence
they are congruences

Well-founded induction

Kleene's fixpoint theorem