Big-step Semantics

Renato Neves

Universidade do Minho

HASLab
HIGH-ASSURANCE
SOFTWARE LABORATORY

Table of Contents

Semantics for every season

Operational semantics	How a program operates
Denotational semantics	What a program is
Axiomatic semantics	Which logical properties a program satisfies

Motivation

The semantics $\longrightarrow^\star$ provides a notion of program equivalence

$$\mathrm{p} \equiv \mathrm{q} \text{ iff } \Big(\langle \mathrm{p}, \sigma \rangle \longrightarrow^\star v \text{ iff } \langle \mathrm{q}, \sigma \rangle \longrightarrow^\star v \Big)$$

Motivation

The semantics $\longrightarrow^\star$ provides a notion of program equivalence

$$\mathrm{p} \equiv \mathrm{q} \text{ iff } \Big(\langle \mathrm{p}, \sigma \rangle \longrightarrow^\star v \text{ iff } \langle \mathrm{q}, \sigma \rangle \longrightarrow^\star v\Big)$$

This leads us to a previous slide . . .

The search for meaning

Examples

- $\mathtt{p}; (\mathtt{q}; \mathtt{r}) \stackrel{?}{=} (\mathtt{p}; \mathtt{q}); \mathtt{r}$
- $\mathtt{p} \parallel \mathtt{q} \stackrel{?}{=} \mathtt{q} \parallel \mathtt{p}$
- $\left(\mathtt{p} +_{\frac{1}{2}} \mathtt{q}\right); \mathtt{r} \stackrel{?}{=} \mathtt{p}; \mathtt{r} +_{\frac{1}{2}} \mathtt{q}; \mathtt{r}$
- $\mathtt{entangle}(\mathtt{x}, \mathtt{y}) \stackrel{?}{=}$ spooky action

Motivation

However (dis)proving equivalences via $\longrightarrow^\star$ is quite cumbersome

Due to equivalence being concerned only with output ... not intermediate steps ...

... and outputs obtained via $\longrightarrow^\star$ relying on these

Motivation

However (dis)proving equivalences via $\longrightarrow^\star$ is quite cumbersome

Due to equivalence being concerned only with output ... not intermediate steps ...

... and outputs obtained via $\longrightarrow^\star$ relying on these

Can we build a more direct, big-step semantics ?

Table of Contents

A simple while-language

Arithmetic expressions

$\mathtt{e} ::= \mathtt{n} \mid \mathtt{e} \cdot \mathtt{e} \mid \mathtt{x} \mid \mathtt{e} + \mathtt{e}$

Programs

$\mathtt{p} ::= \mathtt{x} := \mathtt{e} \mid \mathtt{p} \,\mathtt{;}\, \mathtt{p} \mid \mathtt{if}\ \mathtt{b}\ \mathtt{then}\ \mathtt{p}\ \mathtt{else}\ \mathtt{p} \mid \mathtt{while}\ \mathtt{b}\ \mathtt{do}\ \{\ \mathtt{p}\ \}$

A simple while-language

Arithmetic expressions

$\mathrm{e} ::= \mathrm{n} \mid \mathrm{e} \cdot \mathrm{e} \mid \mathrm{x} \mid \mathrm{e} + \mathrm{e}$

Programs

$\mathrm{p} ::= \mathrm{x} := \mathrm{e} \mid \mathrm{p} \,\texttt{;}\, \mathrm{p} \mid \texttt{if}\ \mathrm{b}\ \texttt{then}\ \mathrm{p}\ \texttt{else}\ \mathrm{p} \mid \texttt{while}\ \mathrm{b}\ \texttt{do}\ \{\ \mathrm{p}\ \}$

Homework: provide semantics to the arithmetic expressions

A while-language and its semantics

$$\frac{\langle \mathtt{e}, \sigma \rangle \Downarrow \mathtt{v}}{\langle \mathtt{x} := \mathtt{e}, \sigma \rangle \Downarrow \sigma[\mathtt{v}/\mathtt{x}]} \ (\mathsf{asg}) \qquad \frac{\langle \mathtt{p}, \sigma \rangle \Downarrow \sigma' \qquad \langle \mathtt{q}, \sigma' \rangle \Downarrow \sigma''}{\langle \mathtt{p} \mathbin{;} \mathtt{q}, \sigma \rangle \Downarrow \sigma''} \ (\mathsf{seq})$$

$$\frac{\langle \mathtt{b}, \sigma \rangle \Downarrow \mathtt{tt} \qquad \langle \mathtt{p}, \sigma \rangle \Downarrow \sigma'}{\langle \mathtt{if}\ \mathtt{b}\ \mathtt{then}\ \mathtt{p}\ \mathtt{else}\ \mathtt{q}, \sigma \rangle \Downarrow \sigma'} \ (\mathsf{if}_1) \qquad \frac{\langle \mathtt{b}, \sigma \rangle \Downarrow \mathtt{ff} \qquad \langle \mathtt{q}, \sigma \rangle \Downarrow \sigma'}{\langle \mathtt{if}\ \mathtt{b}\ \mathtt{then}\ \mathtt{p}\ \mathtt{else}\ \mathtt{q}, \sigma \rangle \Downarrow \sigma'} \ (\mathsf{if}_2)$$

$$\frac{\langle \mathtt{b}, \sigma \rangle \Downarrow \mathtt{tt} \qquad \langle \mathtt{p}, \sigma \rangle \Downarrow \sigma' \qquad \langle \mathtt{while}\ \mathtt{b}\ \mathtt{do}\ \{\ \mathtt{p}\ \}, \sigma' \rangle \Downarrow \sigma''}{\langle \mathtt{while}\ \mathtt{b}\ \mathtt{do}\ \{\ \mathtt{p}\ \}, \sigma \rangle \Downarrow \sigma''} \ (\mathsf{wh}_1)$$

$$\frac{\langle \mathtt{b}, \sigma \rangle \Downarrow \mathtt{ff}}{\langle \mathtt{while}\ \mathtt{b}\ \mathtt{do}\ \{\ \mathtt{p}\ \}, \sigma \rangle \Downarrow \sigma} \ (\mathsf{wh}_2)$$

The semantics at work

Program $\mathtt{x} := \mathtt{x} + 1 \,;\, \mathtt{x} := \mathtt{x} + 2$ corresponds to the syntax tree

$$\begin{tikzcd}[column sep=small] & (\,;\,) \arrow[dl] \arrow[dr] & \\ \mathtt{x} := \mathtt{x} + 1 & & \mathtt{x} := \mathtt{x} + 2 \end{tikzcd}$$

Memory $\sigma = x \mapsto 3$ yields the derivation tree

$$\dfrac{\dfrac{\langle \mathtt{x} + 1, \mathtt{x} \mapsto 3 \rangle \Downarrow 4}{\langle \mathtt{x} := \mathtt{x} + 1, \mathtt{x} \mapsto 3 \rangle \Downarrow \mathtt{x} \mapsto 4} \qquad \dfrac{\langle \mathtt{x} + 2, \mathtt{x} \mapsto 4 \rangle \Downarrow 6}{\langle \mathtt{x} := \mathtt{x} + 2, \mathtt{x} \mapsto 4 \rangle \Downarrow \mathtt{x} \mapsto 6}}{\langle \mathtt{x} := \mathtt{x} + 1 \,;\, \mathtt{x} := \mathtt{x} + 2, \mathtt{x} \mapsto 3 \rangle \Downarrow \mathtt{x} \mapsto 6}$$

Exercise

Provide a big-step semantics to the propositional language

$$\mathtt{b} ::= \mathtt{x} \mid \mathtt{b} \wedge \mathtt{b} \mid \neg \mathtt{b}$$

Equivalence of while-programs

The previous semantics yields the following notion of equivalence

$\mathtt{p} \equiv \mathtt{q}$ if for all environments σ

$$\langle \mathtt{p}, \sigma \rangle \Downarrow \sigma' \text{ iff } \langle \mathtt{q}, \sigma \rangle \Downarrow \sigma'$$

Examples of equivalent terms

- $(\mathtt{p} \mathbin{;} \mathtt{q}) \mathbin{;} \mathtt{r} \equiv \mathtt{p} \mathbin{;} (\mathtt{q} \mathbin{;} \mathtt{r})$
- $(\mathtt{if}\ \mathtt{b}\ \mathtt{then}\ \mathtt{p}\ \mathtt{else}\ \mathtt{q}) \mathbin{;} \mathtt{r} \equiv \mathtt{if}\ \mathtt{b}\ \mathtt{then}\ \mathtt{p} \mathbin{;} \mathtt{r}\ \mathtt{else}\ \mathtt{q} \mathbin{;} \mathtt{r}$

The relation to the small-step semantics

Lemma

$$\langle \mathrm{p}, \sigma \rangle \longrightarrow \langle \mathrm{p}', \sigma' \rangle \Downarrow \sigma'' \textit{ implies } \langle \mathrm{p}, \sigma \rangle \Downarrow \sigma''$$

Proof.

Induction over the rules concerning $\longrightarrow$ □

The relation to the small-step semantics

Lemma

$$\langle \mathrm{p}, \sigma \rangle \longrightarrow \langle \mathrm{p}', \sigma' \rangle \Downarrow \sigma'' \textit{ implies } \langle \mathrm{p}, \sigma \rangle \Downarrow \sigma''$$

Proof.

Induction over the rules concerning $\longrightarrow$ $\square$

Theorem

$$\langle \mathrm{p}, \sigma \rangle \longrightarrow^{\star} \sigma' \textit{ iff } \langle \mathrm{p}, \sigma \rangle \Downarrow \sigma'$$

Proof.

Left-to-right-direction: previous lemma and induction over the rules concerning $\longrightarrow^{\star}$

Right-to-left direction: induction over the rules concerning $\Downarrow$ $\square$