

CS 4110

Programming Languages & Logics

Lecture 5
The IMP Language



Simple imperative language

We'll now consider a more realistic programming language...

Simple imperative language

We'll now consider a more realistic programming language...

arithmetic expressions $a \in \mathbf{Aexp}$ $a ::= x \mid n \mid a_1 + a_2 \mid a_1 \times a_2$

Simple imperative language

We'll now consider a more realistic programming language...

arithmetic expressions $a \in \mathbf{Aexp}$ $a ::= x \mid n \mid a_1 + a_2 \mid a_1 \times a_2$

Boolean expressions $b \in \mathbf{Bexp}$ $b ::= \mathbf{true} \mid \mathbf{false} \mid a_1 < a_2$

Simple imperative language

We'll now consider a more realistic programming language...

arithmetic expressions $a \in \mathbf{Aexp}$ $a ::= x \mid n \mid a_1 + a_2 \mid a_1 \times a_2$
Boolean expressions $b \in \mathbf{Bexp}$ $b ::= \mathbf{true} \mid \mathbf{false} \mid a_1 < a_2$
commands $c \in \mathbf{Com}$ $c ::= \mathbf{skip}$
| $x := a$
| $c_1; c_2$
| **if** b **then** c_1 **else** c_2
| **while** b **do** c

Small-Step Semantics

Three relations, one for each syntactic category:

$$\rightarrow_{\mathbf{Aexp}} \subseteq (\mathbf{Store} \times \mathbf{Aexp}) \times (\mathbf{Store} \times \mathbf{Aexp})$$

$$\rightarrow_{\mathbf{Bexp}} \subseteq (\mathbf{Store} \times \mathbf{Bexp}) \times (\mathbf{Store} \times \mathbf{Bexp})$$

$$\rightarrow_{\mathbf{Com}} \subseteq (\mathbf{Store} \times \mathbf{Com}) \times (\mathbf{Store} \times \mathbf{Com})$$

We'll typically just use \rightarrow where the specific relation we mean is clear from context.

Small-Step Semantics

For example:

$\langle \sigma, \mathbf{if\ true\ then\ } x := 1 \mathbf{\ else\ } x := 2 \rangle$

Small-Step Semantics

For example:

$$\begin{aligned} & \langle \sigma, \mathbf{if\ true\ then\ } x := 1 \mathbf{\ else\ } x := 2 \rangle \\ \rightarrow_{\mathbf{com}} & \langle \sigma, x := 1 \rangle \end{aligned}$$

Small-Step Semantics

For example:

$$\begin{aligned} & \langle \sigma, \mathbf{if\ true\ then\ } x := 1 \mathbf{\ else\ } x := 2 \rangle \\ \rightarrow_{\mathbf{com}} & \langle \sigma, x := 1 \rangle \\ \rightarrow_{\mathbf{com}} & \langle \sigma[x \mapsto 1], \mathbf{skip} \rangle \end{aligned}$$

Small-Step Semantics

Arithmetic expressions:

$$\frac{n = \sigma(x)}{\langle \sigma, x \rangle \rightarrow \langle \sigma, n \rangle}$$

Small-Step Semantics

Arithmetic expressions:

$$\frac{\langle \sigma, a_1 \rangle \rightarrow \langle \sigma, a'_1 \rangle}{\langle \sigma, a_1 + a_2 \rangle \rightarrow \langle \sigma, a'_1 + a_2 \rangle}$$

$$\frac{\langle \sigma, a_2 \rangle \rightarrow \langle \sigma, a'_2 \rangle}{\langle \sigma, n + a_2 \rangle \rightarrow \langle \sigma, n + a'_2 \rangle}$$

$$\frac{p = n + m}{\langle \sigma, n + m \rangle \rightarrow \langle \sigma, p \rangle}$$

Small-Step Semantics

Arithmetic expressions:

$$\frac{\langle \sigma, a_1 \rangle \rightarrow \langle \sigma, a'_1 \rangle}{\langle \sigma, a_1 \times a_2 \rangle \rightarrow \langle \sigma, a'_1 \times a_2 \rangle}$$

$$\frac{\langle \sigma, a_2 \rangle \rightarrow \langle \sigma, a'_2 \rangle}{\langle \sigma, n \times a_2 \rangle \rightarrow \langle \sigma, n \times a'_2 \rangle}$$

$$\frac{p = n \times m}{\langle \sigma, n \times m \rangle \rightarrow \langle \sigma, p \rangle}$$

Small-Step Semantics

Boolean expressions:

$$\frac{\langle \sigma, a_1 \rangle \rightarrow \langle \sigma, a'_1 \rangle}{\langle \sigma, a_1 < a_2 \rangle \rightarrow \langle \sigma, a'_1 < a_2 \rangle}$$

$$\frac{\langle \sigma, a_2 \rangle \rightarrow \langle \sigma, a'_2 \rangle}{\langle \sigma, n < a_2 \rangle \rightarrow \langle \sigma, n < a'_2 \rangle}$$

$$\frac{n < m}{\langle \sigma, n < m \rangle \rightarrow \langle \sigma, \mathbf{true} \rangle}$$

$$\frac{n \geq m}{\langle \sigma, n < m \rangle \rightarrow \langle \sigma, \mathbf{false} \rangle}$$

Small-Step Semantics

Commands:

$$\frac{\langle \sigma, a \rangle \rightarrow \langle \sigma, a' \rangle}{\langle \sigma, x := a \rangle \rightarrow \langle \sigma, x := a' \rangle}$$

$$\frac{}{\langle \sigma, x := n \rangle \rightarrow \langle \sigma[x \mapsto n], \mathbf{skip} \rangle}$$

Small-Step Semantics

Commands:

$$\frac{\langle \sigma, c_1 \rangle \rightarrow \langle \sigma', c'_1 \rangle}{\langle \sigma, c_1; c_2 \rangle \rightarrow \langle \sigma', c'_1; c_2 \rangle}$$

$$\frac{}{\langle \sigma, \mathbf{skip}; c_2 \rangle \rightarrow \langle \sigma, c_2 \rangle}$$

Small-Step Semantics

Commands:

$$\frac{\langle \sigma, b \rangle \rightarrow \langle \sigma, b' \rangle}{\langle \sigma, \mathbf{if } b \mathbf{ then } c_1 \mathbf{ else } c_2 \rangle \rightarrow \langle \sigma, \mathbf{if } b' \mathbf{ then } c_1 \mathbf{ else } c_2 \rangle}$$

$$\frac{}{\langle \sigma, \mathbf{if true then } c_1 \mathbf{ else } c_2 \rangle \rightarrow \langle \sigma, c_1 \rangle}$$

$$\frac{}{\langle \sigma, \mathbf{if false then } c_1 \mathbf{ else } c_2 \rangle \rightarrow \langle \sigma, c_2 \rangle}$$

Small-Step Semantics

Commands:

$$\frac{}{\langle \sigma, \mathbf{while\ } b \mathbf{ do\ } c \rangle \rightarrow \langle \sigma, \mathbf{if\ } b \mathbf{ then\ } (c; \mathbf{while\ } b \mathbf{ do\ } c) \mathbf{ else\ skip} \rangle}$$

Large-Step Semantics

Again three relations, one for each syntactic category:

$$\Downarrow_{\mathbf{Aexp}} \subseteq (\mathbf{Store} \times \mathbf{Aexp}) \times \mathbf{Int}$$

$$\Downarrow_{\mathbf{Bexp}} \subseteq (\mathbf{Store} \times \mathbf{Bexp}) \times \mathbf{Bool}$$

$$\Downarrow_{\mathbf{Com}} \subseteq (\mathbf{Store} \times \mathbf{Com}) \times \mathbf{Store}$$

And again, we'll typically just use \Downarrow where the specific relation we mean is clear from context.

Large-Step Semantics

$$\frac{}{\langle \sigma, n \rangle \Downarrow n} \qquad \frac{\sigma(x) = n}{\langle \sigma, x \rangle \Downarrow n}$$

$$\frac{\langle \sigma, e_1 \rangle \Downarrow n_1 \quad \langle \sigma, e_2 \rangle \Downarrow n_2 \quad n = n_1 + n_2}{\langle \sigma, e_1 + e_2 \rangle \Downarrow n}$$

$$\frac{\langle \sigma, e_1 \rangle \Downarrow n_1 \quad \langle \sigma, e_2 \rangle \Downarrow n_2 \quad n = n_1 \times n_2}{\langle \sigma, e_1 \times e_2 \rangle \Downarrow n}$$

Large-Step Semantics

$$\overline{\langle \sigma, \mathbf{true} \rangle} \Downarrow \mathbf{true}$$
$$\overline{\langle \sigma, \mathbf{false} \rangle} \Downarrow \mathbf{false}$$
$$\frac{\langle \sigma, a_1 \rangle \Downarrow n_1 \quad \langle \sigma, a_2 \rangle \Downarrow n_2 \quad n_1 < n_2}{\langle \sigma, a_1 < a_2 \rangle \Downarrow \mathbf{true}}$$
$$\frac{\langle \sigma, a_1 \rangle \Downarrow n_1 \quad \langle \sigma, a_2 \rangle \Downarrow n_2 \quad n_1 \geq n_2}{\langle \sigma, a_1 < a_2 \rangle \Downarrow \mathbf{false}}$$

Large-Step Semantics

SKIP

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

Large-Step Semantics

$$\text{ASSGN} \frac{\langle \sigma, a \rangle \Downarrow n}{\langle \sigma, x := a \rangle \Downarrow \sigma[x \mapsto n]}$$

Large-Step Semantics

$$\text{SEQ} \frac{\langle \sigma, c_1 \rangle \Downarrow \sigma' \quad \langle \sigma', c_2 \rangle \Downarrow \sigma''}{\langle \sigma, c_1; c_2 \rangle \Downarrow \sigma''}$$

Large-Step Semantics

IF-T

$$\frac{\langle \sigma, b \rangle \Downarrow \mathbf{true} \quad \langle \sigma, c_1 \rangle \Downarrow \sigma'}{\langle \sigma, \mathbf{if } b \mathbf{ then } c_1 \mathbf{ else } c_2 \rangle \Downarrow \sigma'}$$

Large-Step Semantics

IF-T

$$\frac{\langle \sigma, b \rangle \Downarrow \mathbf{true} \quad \langle \sigma, c_1 \rangle \Downarrow \sigma'}{\langle \sigma, \mathbf{if } b \mathbf{ then } c_1 \mathbf{ else } c_2 \rangle \Downarrow \sigma'}$$

IF-F

$$\frac{\langle \sigma, b \rangle \Downarrow \mathbf{false} \quad \langle \sigma, c_2 \rangle \Downarrow \sigma'}{\langle \sigma, \mathbf{if } b \mathbf{ then } c_1 \mathbf{ else } c_2 \rangle \Downarrow \sigma'}$$

Large-Step Semantics

$$\begin{array}{c} \text{WHILE-F} \\ \langle \sigma, b \rangle \Downarrow \mathbf{false} \\ \hline \langle \sigma, \mathbf{while } b \mathbf{ do } c \rangle \Downarrow \sigma \end{array}$$

Large-Step Semantics

WHILE-F

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

WHILE-T

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