CS101.3

Programing Language Semantics

Lecture 1

January 9, 2004

Quick Information

Time:	F 14:00 – 15:00
Place:	Jorgensen 74
Instructors:	Jason Hickey and Aleksey Nogin
Office Hours:	Jorgensen 60; TBA, and by appt
Units:	4 (1+0+3), pass/fail or letter grade
Course Home Page:	`http://nogin.org/cs101/`
Admin email:	`cs101-admin@metaprl.org`
Mailing list:	`cs101-class@metaprl.org`
Textbook:	Glynn Winskel. The Formal Semantics of Programming Languages. An Introduction.

Assigning Meaning to Programs

- How the program executes — *operational semantics*
- What the program as a whole *means* — *denotational semantics*
- What properties does a program have — *axiomatic semantics*

Assigning Meaning to Programs — Example

```
let fact i =
    if i = 1 then 1 else fact (i-1) * i
```

- How the program executes (*operational semantics*): “to compute `fact`, first compare input to 1, if equals, then return 1, else . . .”;
- What the program as a whole *means* (*denotational semantics*): “`fact` computes the factorial function”;
- What properties does a program have (*axiomatics semantics*): “when input is a positive integer, `fact` would terminate”.

IMP— a Simple Imperative Language

- Numbers (**N**): m, n
- Locations (**Loc**): X, Y
- Arithmetical expressions (**Aexp**):
  $a ::= n \mid X \mid a_1 + a_2 \mid a_1 - a_2 \mid a_1 * a_2$
- Boolean expressions (**Bexp**):
  $b ::= \mathbf{true} \mid \mathbf{false} \mid a_1 = a_2 \mid a_1 \leq a_2 \mid$
  $\neg b \mid b_1 \vee b_2 \mid b_1 \wedge b_2$
- Commands (**Com**):
  $c ::= \mathbf{skip} \mid X := a \mid c_1; c_2 \mid \mathbf{if}\ b\ \mathbf{then}\ c_1\ \mathbf{else}\ c_2 \mid$
  $\mathbf{while}\ b\ \mathbf{do}\ c$

Evaluation: State

A *state* σ maps locations to numbers. $\Sigma = \mathbf{Loc} \to \mathbf{N}$ — set all all states. $\sigma(X)$ — contents of location X in state σ.

For $\sigma \in \Sigma$ and $a \in \mathbf{Aexp}$ we will define a relation $\langle \sigma, a \rangle \to n$:

- $\langle \sigma, n \rangle \to n$
- $\langle \sigma, X \rangle \to \sigma(X)$
- $\dfrac{\langle \sigma, a_1 \rangle \to n_1 \quad \langle \sigma, a_2 \rangle \to n_2}{\langle \sigma, a_1 + a_2 \rangle \to n}$, where $n = n_1 + n_2$.
- $\dfrac{\langle \sigma, a_1 \rangle \to n_1 \quad \langle \sigma, a_2 \rangle \to n_2}{\langle \sigma, a_1 - a_2 \rangle \to n}$, where $n = n_1 - n_2$.
- *etc*

Evaluation Example

In σ_0 that maps all locations to 0, evaluate $(I+1)*(4-2)$

$$\dfrac{\dfrac{\dfrac{}{\langle \sigma_0, I\rangle \rightarrow 0} \quad \dfrac{}{\langle \sigma_0, 1\rangle \rightarrow 1}}{\langle \sigma_0, I+1\rangle \rightarrow 1} \qquad \dfrac{\dfrac{}{\langle \sigma_0, 4\rangle \rightarrow 4} \quad \dfrac{}{\langle \sigma_0, 2\rangle \rightarrow 2}}{\langle \sigma_0, 4-2\rangle \rightarrow 2}}{\langle \sigma_0, (I+1)*(4-2)\rangle \rightarrow 2}$$

Evaluation of Booleans

- $\langle\sigma, \texttt{true}\rangle \rightarrow \texttt{true}$ and $\langle\sigma, \texttt{false}\rangle \rightarrow \texttt{false}$
- $\dfrac{\langle\sigma, a_1\rangle \rightarrow n_1 \quad \langle\sigma, a_2\rangle \rightarrow n_2}{\langle\sigma, a_1 = a_2\rangle \rightarrow \texttt{true}}$, if n_1 and n_2 are equal;
- $\dfrac{\langle\sigma, a_1\rangle \rightarrow n_1 \quad \langle\sigma, a_2\rangle \rightarrow n_2}{\langle\sigma, a_1 = a_2\rangle \rightarrow \texttt{false}}$, if n_1 and n_2 are unequal;
- $\dfrac{\langle\sigma, b\rangle \rightarrow \texttt{true}}{\langle\sigma, \neg b\rangle \rightarrow \texttt{false}}$ and $\dfrac{\langle\sigma, b\rangle \rightarrow \texttt{false}}{\langle\sigma, \neg b\rangle \rightarrow \texttt{true}}$
- *etc*

Equivalence of Expressions

Two expressions are equivalent if they evaluate to the same thing in any state:

$$a_1 \sim a_2 \textit{ iff } \forall \sigma \in \Sigma . \forall n \in \mathbf{N}.\ \langle \sigma, a_1 \rangle \ \rightarrow \ n \Leftrightarrow \langle \sigma, a_2 \rangle \ \rightarrow \ n$$

Similarly, for booleans:

$$b_1 \sim b_2 \textit{ iff}$$
$$\forall \sigma \in \Sigma . \forall t \in \{\mathtt{true}, \mathtt{false}\}.\ \langle \sigma, b_1 \rangle \ \rightarrow \ t \Leftrightarrow \langle \sigma, b_2 \rangle \ \rightarrow \ t$$

Example. Whenever $b_1 \sim \mathtt{true}$, $b_1 \vee b_2 \sim \mathtt{true}$ (for any $b_2 \in \mathtt{Bexp}$).

Evaluation of Commands

- $\langle \sigma, \texttt{skip} \rangle \rightarrow \sigma \qquad \dfrac{\langle \sigma, a \rangle \rightarrow n}{\langle \sigma, X := a \rangle \rightarrow \sigma[n/X]}$

- $$\frac{\langle \sigma, c_1 \rangle \rightarrow \sigma' \quad \langle \sigma', c_2 \rangle \rightarrow \sigma''}{\langle \sigma, c_1; c_2 \rangle \rightarrow \sigma''}$$

- $$\frac{\langle \sigma, b \rangle \rightarrow \texttt{true} \quad \langle \sigma, c_1 \rangle \rightarrow \sigma'}{\langle \sigma, \texttt{if}\ b\ \texttt{then}\ c_1\ \texttt{else}\ c_2 \rangle \rightarrow \sigma'}$$
  $$\frac{\langle \sigma, b \rangle \rightarrow \texttt{false} \quad \langle \sigma, c_2 \rangle \rightarrow \sigma'}{\langle \sigma, \texttt{if}\ b\ \texttt{then}\ c_1\ \texttt{else}\ c_2 \rangle \rightarrow \sigma'}$$

- $$\frac{\langle \sigma, b \rangle \rightarrow \texttt{false}}{\langle \sigma, \texttt{while}\ b\ \texttt{do}\ c \rangle \rightarrow \sigma}$$
  $$\frac{\langle \sigma, b \rangle \rightarrow \texttt{true} \quad \langle \sigma, c \rangle \rightarrow \sigma' \quad \langle \sigma', \texttt{while}\ b\ \texttt{do}\ c \rangle \rightarrow \sigma''}{\langle \sigma, \texttt{while}\ b\ \texttt{do}\ c \rangle \rightarrow \sigma''}$$

Theorem

Let w := **`while`** b **`do`** c. Then

$$w \sim \texttt{if}\ b\ \texttt{then}\ c; w\ \texttt{else skip}$$