

Semântica Formal
Trabalho Final:
Linguagem de Programação *Esquema*

Alex Gliesch
Daniel Silva
Hyago Sallet

2 de Dezembro de 2013

A linguagem

- Funcional
- Baseada em Scheme, mas com tipagem estática
- Muitos parênteses
- Fácil de parsear

Principais elementos

- *empty*
- $(\textit{lambda} : T_r (x_i : T_i \dots) e)$
- $(\textit{if} \textit{teste} e_1 e_2)$
- $(\textit{define} x e)$
- $(\textit{local} (e_i \dots) e_r)$
- $(\textit{begin} e_i \dots)$
- $(\textit{list} e_i \dots)$
- $+, -, *, <, \textit{and}, \textit{not}$ e *empty*?

Tipos

- int (precisão arbitrária)
- bool
- string
- [] (lista vazia)
- composição entre eles (funções, listas)

$t ::=$

- (t)
- n
- b
- s
- $empty$
- $(\lambda x_1 : T_1 \ x_2 : T_2 \ \dots \ x_n : T_n) \ t_r$
- $(if \ t_1 \ t_2 \ t_3)$
- $(define \ x \ t_1)$
- $(local \ (t_1 \ t_2 \ \dots \ t_n) \ t_r)$
- $(binop \ t_1 \ t_2)$
- $(unop \ t_1)$
- $(n\text{-ary} \ t_1 \ t_2 \ \dots \ t_n)$

$n \in \mathbb{Z}$

$b \in \{true, false\}$

$s \in$ conjunto das strings possíveis

$binop \in \{+, -, *, and, cons\}$

$unop \in \{not, empty?, first, rest\}$

$n\text{-ary} \in \{begin, list\}$

x representa um identificador, definido pelo usuário

Demonstração

Executar o interpretador.

Semântica operacional - contexto

- α : mapeia um identificador a um valor
- Representa o contexto atual

$$\frac{x \in Dom(\alpha) \text{ e } \alpha(x) = v}{\langle \alpha, x \rangle \rightarrow \langle \alpha, v \rangle} \quad [CTX]$$

Semântica operacional - + e and

$$\frac{[[n]] = [[n_1 + n_2]]}{\langle \alpha, (+ \ n_1 \ n_2) \rangle \rightarrow \langle \alpha, n \rangle} \quad [OP+]$$

$$\frac{[[b]] = [[b_1 \text{ AND } b_2]]}{\langle \alpha, (\text{and } b_1 \ b_2) \rangle \rightarrow \langle \alpha, b \rangle} \quad [AND]$$

Semântica operacional - cons

$$\frac{l_1 = (v_i^{i \in 1..n}) \quad n \geq 1}{\langle \alpha, (\text{cons } v_0 \ l_1) \rangle \rightarrow \langle \alpha, (v_i^{i \in 0..n}) \rangle} \quad [\text{CONS1}]$$

$$\frac{}{\langle \alpha, (\text{cons } v_1 \ \text{empty}) \rangle \rightarrow \langle \alpha, (v_1) \rangle} \quad [\text{CONS2}]$$

Semântica operacional - first

$$\frac{l_1 = (t_i^{i \in 1..n}) \quad n \geq 1}{\langle \alpha, (\text{first } l_1) \rangle \rightarrow \langle \alpha, t_1 \rangle} \quad [FIRST1]$$

$$\frac{}{\langle \alpha, (\text{first empty}) \rangle \rightarrow \langle \alpha, \text{empty} \rangle} \quad [FIRST2]$$

Semântica operacional - rest

$$\frac{l_1 = (t_i^{i \in 1..n}) \quad n > 1}{\langle \alpha, (\text{rest } l_1) \rangle \rightarrow \langle \alpha, (t_i^{i \in 2..n}) \rangle} \quad [\text{REST1}]$$

$$\frac{}{\langle \alpha, (\text{rest empty}) \rangle \rightarrow \langle \alpha, \text{empty} \rangle} \quad [\text{REST2}]$$

$$\frac{l_1 = (t_1)}{\langle \alpha, (\text{rest } l_1) \rangle \rightarrow \langle \alpha, \text{empty} \rangle} \quad [\text{REST3}]$$

Semântica operacional - if

$$\frac{}{\langle \alpha, (\text{if } \text{true } t_1 \ t_2) \rangle \rightarrow \langle \alpha, t_1 \rangle} \quad [IF1]$$

$$\frac{}{\langle \alpha, (\text{if } \text{false } t_1 \ t_2) \rangle \rightarrow \langle \alpha, t_2 \rangle} \quad [IF2]$$

$$\frac{t_1 \rightarrow t'_1}{\langle \alpha, (\text{if } t_1 \ t_2 \ t_3) \rangle \rightarrow \langle \alpha', (\text{if } t'_1 \ t_2 \ t_3) \rangle} \quad [IF3]$$

Semântica operacional - aplicação de função

$$\frac{}{\langle \alpha, ((\textit{lambda}: T_r (x_i: T_i^{i \in 1..n}) t_r) v_i^{i \in 1..n}) \rangle \rightarrow \langle \alpha, \{v_i/x_i\}^{i \in 1..n} t_r \rangle} \quad [\textit{APP1}]$$

Semântica operacional - define

$$\frac{t_1 \rightarrow t'_1}{\langle \alpha, (\text{define } x \ t_1) \rangle \rightarrow \langle \alpha', (\text{define } x \ t'_1) \rangle} \quad [\text{DEFINE1}]$$

$$\frac{}{\langle \alpha, (\text{define } x \ v_1) \rangle \rightarrow \langle \alpha, [x \mapsto v_1], v_1 \rangle} \quad [\text{DEFINE2}]$$

Sistema de tipos

- Γ : mapeamento de identificadores para tipos
- $T ::= int \mid bool \mid string \mid [] \mid T, \dots, T \rightarrow T \mid [T]$

$$\frac{\Gamma(x) = T}{\Gamma \vdash x : T} \quad [TVAR]$$

Sistema de tipos - tipos das operações

$$\frac{\Gamma \vdash e_1 : int \quad \Gamma \vdash e_2 : int}{\Gamma \vdash (+ e_1 e_2) : int} \quad [TOP+]$$

$$\frac{\Gamma \vdash e_1 : bool \quad \Gamma \vdash e_2 : bool}{\Gamma \vdash (and e_1 e_2) : bool} \quad [TAND]$$

Sistema de tipos - cons

$$\frac{\Gamma \vdash e_1 : T \quad \Gamma \vdash e_2 : [T]}{\Gamma \vdash (\text{cons } e_1 \ e_2) : [T]} \quad [TCONS1]$$

$$\frac{\Gamma \vdash e_1 : T}{\Gamma \vdash (\text{cons } e_1 \ \text{empty}) : [T]} \quad [TCONS2]$$

Sistema de tipos - first

$$\frac{\Gamma \vdash e_1 : [T]}{\Gamma \vdash (\text{first } e_1) : T} \quad [TFIRST1]$$

$$\frac{}{\Gamma \vdash (\text{first empty}) : []} \quad [TFIRST2]$$

Sistema de tipos - if e begin

$$\frac{\Gamma \vdash e_1 : \text{bool} \quad e_2 : T \quad e_3 : T}{\Gamma \vdash (\text{if } e_1 \ e_2 \ e_3) : T} \quad [TIF]$$

$$\frac{\Gamma \vdash e_n : T \quad n \geq 1}{\Gamma \vdash (\text{begin } e_i^{i \in 1..n}) : T} \quad [TBEGIN]$$

Sistema de tipos - define e aplicação

$$\frac{\Gamma \vdash e_1 : T}{\Gamma \vdash (\text{define } x \ e_1) : T} \quad [TDEFINE]$$

$$\frac{\Gamma \vdash e_i : T_i \quad \forall i \in 1..n}{\Gamma \vdash ((\text{lambda} : T_r \ (x_i : T_i^{i \in 1..n}) \ e_r) \ e_i^{i \in 1..n}) : T_r} \quad [TAPP]$$