Linguaggi di Programmazione

Roberta Gori

# Semantica operazionale di HOFL-cap 7.2

# Attenzione

$$
\begin{array}{rcl}
t & ::= & x \mid n \mid t_0 \ \mathsf{op}\ t_1 \mid \textbf{if } t \textbf{ then } t_0 \textbf{ else } t_1 \\
 & \mid & (t_0, t_1) \mid \textbf{fst}(t) \mid \textbf{snd}(t) \\
 & \mid & \lambda x.\ t \mid t_0\ t_1 \\
 & \mid & \textbf{rec } x.\ t
\end{array}
$$

$$\tau \ ::= \ int \mid \tau_0 * \tau_1 \mid \tau_0 \to \tau_1$$

$$\frac{}{x : \widehat{x}} \qquad \frac{}{n : int} \qquad \frac{t_0 : int \quad t_1 : int}{t_0 \ \mathsf{op}\ t_1 : int} \qquad \frac{t : int \quad t_0 : \tau \quad t_1 : \tau}{\textbf{if } t \textbf{ then } t_0 \textbf{ else } t_1 : \tau}$$

$$\frac{t_0 : \tau_0 \quad t_1 : \tau_1}{(t_0, t_1) : \tau_0 * \tau_1} \qquad \frac{t : \tau_0 * \tau_1}{\textbf{fst}(t) : \tau_0} \qquad \frac{t : \tau_0 * \tau_1}{\textbf{snd}(t) : \tau_1}$$

$$\frac{x : \tau_0 \quad t : \tau_1}{\lambda x.\ t : \tau_0 \to \tau_1} \qquad \frac{t_1 : \tau_0 \to \tau_1 \quad t_0 : \tau_0}{t_1\ t_0 : \tau_1}$$

$$\frac{x : \tau \quad t : \tau}{\textbf{rec } x.\ t : \tau}$$

assegniamo una semantica
solo ai termini che sono:
ben formati e chiusi

$$t : \tau \qquad\qquad \mathrm{fv}(t) = \varnothing$$

# Forme Canoniche

# Statements

si legge "si riduce a"

$$t \rightarrow c$$

un termine chiuso

una forma canonica
(un tipo speciale di termine)

semantica operazionale Big step

riduzione alla forma canonica
(attraverso la manipulazione di termini)

# Forme canoniche

insieme di forme canoniche con tipo $\tau$ $\quad C_\tau \subseteq T_\tau$

(laziness)
non e' richiesto
siano in forma canonica

$t$ non e' necessariamente
termine chiuso

$$\frac{}{n \in C_{int}} \qquad \frac{t_0 : \tau_0 \quad t_1 : \tau_1 \quad t_0, t_1 \text{ closed}}{(t_0, t_1) \in C_{\tau_0 * \tau_1}} \qquad \frac{\lambda x.\, t : \tau_0 \to \tau_1 \quad \lambda x.\, t \text{ closed}}{\lambda x.\, t \in C_{\tau_0 \to \tau_1}}$$

# Canonical forms?

$$\frac{}{n \in C_{int}} \qquad \frac{t_0 : \tau_0 \quad t_1 : \tau_1 \quad t_0, t_1 \text{ closed}}{(t_0, t_1) \in C_{\tau_0 * \tau_1}} \qquad \frac{\lambda x.\, t : \tau_0 \to \tau_1 \quad \lambda x.\, t \text{ closed}}{\lambda x.\, t \in C_{\tau_0 \to \tau_1}}$$

| | | | |
|---|---|---|---|
| $1 + 2 \times 3$ | ✗ | $\textbf{if } 0 \textbf{ then } 0 \textbf{ else } 0$ | ✗ |
| $(1, 2)$ | ✓ | $\lambda x.\ 1$ | ✓ |
| $(1 + 2, 2 - 1)$ | ✓ | $\lambda x.\ 1 + 2 \times 3$ | ✓ |
| $\textbf{fst}(1, 2)$ | ✗ | $\lambda x.\ \textbf{fst}(1, 2)$ | ✓ |

# HOFL
# Semantica operazionale Lazy

# Operational semantics: axioms

$$\frac{c \in C_\tau}{c \to c}$$

i.e., expanding the various cases

$$\frac{}{n \to n} \qquad \frac{t_0 : \tau_0 \quad t_1 : \tau_1 \quad t_0, t_1 \text{ closed}}{(t_0, t_1) \to (t_0, t_1)} \qquad \frac{\lambda x.\, t : \tau_0 \to \tau_1 \quad \lambda x.t \text{ closed}}{\lambda x.\, t \to \lambda x.\, t}$$

integers, pairs and abstractions
are already in canonical form

# Semantica op. Lazy

$$\frac{}{n \to n} \qquad \frac{t_0 : \tau_0 \quad t_1 : \tau_1 \quad t_0, t_1 \text{ closed}}{(t_0, t_1) \to (t_0, t_1)} \qquad \frac{\lambda x.\, t : \tau_0 \to \tau_1 \quad \lambda x.t \text{ closed}}{\lambda x.\, t \to \lambda x.\, t}$$

$$\frac{t_0 \to n_0 \quad t_1 \to n_1}{t_0 \text{ op } t_1 \to n_0 \,\underline{\text{op}}\, n_1} \qquad \frac{t \to 0 \quad t_0 \to c_0}{\textbf{if } t \textbf{ then } t_0 \textbf{ else } t_1 \to c_0} \qquad \frac{t \to (t_0, t_1) \quad t_0 \to c_0}{\textbf{fst}(t) \to c_0}$$

$$\frac{t[^{\textbf{rec } x.\, t}/_x] \to c}{\textbf{rec } x.\, t \to c} \qquad \frac{t \to n \quad n \neq 0 \quad t_1 \to c_1}{\textbf{if } t \textbf{ then } t_0 \textbf{ else } t_1 \to c_1} \qquad \frac{t \to (t_0, t_1) \quad t_1 \to c_1}{\textbf{snd}(t) \to c_1}$$

$$\frac{t_1 \to \lambda x.\, t_1' \quad t_1'[^{t_0}/_x] \to c}{(t_1\ t_0) \to c} \ \text{(lazy)}$$

# Sistema di tipi (remind)

$$\frac{}{x : \widehat{x}} \qquad \frac{}{n : int} \qquad \frac{t_0 : int \quad t_1 : int}{t_0 \ \mathsf{op}\ t_1 : int} \qquad \frac{t : int \quad t_0 : \tau \quad t_1 : \tau}{\mathbf{if}\ t\ \mathbf{then}\ t_0\ \mathbf{else}\ t_1 : \tau}$$

$$\frac{t_0 : \tau_0 \quad t_1 : \tau_1}{(t_0, t_1) : \tau_0 * \tau_1} \qquad \frac{t : \tau_0 * \tau_1}{\mathbf{fst}(t) : \tau_0} \qquad \frac{t : \tau_0 * \tau_1}{\mathbf{snd}(t) : \tau_1}$$

$$\frac{x : \tau_0 \quad t : \tau_1}{\lambda x.\ t : \tau_0 \to \tau_1} \qquad \frac{t_1 : \tau_0 \to \tau_1 \quad t_0 : \tau_0}{t_1\ t_0 : \tau_1}$$

$$\frac{x : \tau \quad t : \tau}{\mathbf{rec}\ x.\ t : \tau}$$

# Esempio

$$t \triangleq \underbrace{\lambda \underbrace{x}_{int}. \underbrace{\underbrace{x}_{int} + \underbrace{1}_{int}}_{int}}_{int \to int} \quad : int \to int$$

$$\lambda x.\ x + 1 \to c \ \nwarrow_{c = \lambda x.\ x+1} \square$$

# Esempio

$$t \triangleq \underbrace{\underbrace{(\lambda x.\ x+1}_{int \to int}, \underbrace{\underbrace{1}_{int} + \underbrace{2}_{int}}_{int})}_{(int \to int) * int} : (int \to int) * int$$

$$(\lambda x.\ x+1, 1+2) \to c \ \ \nwarrow_{c=(\lambda x.\ x+1,1+2)} \ \square$$

laziness:
non dobbiamo valutare 1+2

# Esempio

$$t \triangleq \underbrace{\lambda \underbrace{x}_{int * int}.\ \underbrace{\textbf{if } \underbrace{\textbf{fst}(\underbrace{x}_{int * \tau_1})}_{int} \textbf{ then } \underbrace{1}_{int} \textbf{ else } \underbrace{\textbf{snd}(\underbrace{x}_{int * \tau_1})}_{int = \tau_1}}_{int}}_{(int * int) \to int} \quad : (int * int) \to int$$

# Esempio (con.)

$$t \triangleq \lambda x.\ \mathbf{if}\ \mathbf{fst}(x)\ \mathbf{then}\ 1\ \mathbf{else}\ \mathbf{snd}(x)$$

$$t\ (1,2) \to c\ \nwarrow\ t \to \lambda x'.\, t'\ ,\ t'[^{(1,2)}/_{x'}] \to c$$

$$\nwarrow_{x'=x,\ t'=\mathbf{if}\ \ldots}\ (\mathbf{if}\ \mathbf{fst}(x)\ \mathbf{then}\ 1\ \mathbf{else}\ \mathbf{snd}(x))[^{(1,2)}/_{x}] \to c$$

$$= \mathbf{if}\ \mathbf{fst}(1,2)\ \mathbf{then}\ 1\ \mathbf{else}\ \mathbf{snd}(1,2) \to c$$

$$\nwarrow\ \mathbf{fst}(1,2) \to n\ ,\ n \neq 0\ ,\ \mathbf{snd}(1,2) \to c$$

$$\nwarrow\ (1,2) \to (n_1, n_2)\ ,\ n_1 \to n\ ,\ n \neq 0\ ,\ \mathbf{snd}(1,2) \to c$$

$$\nwarrow^{*}_{n_1=1, n_2=2, n=1}\ \mathbf{snd}(1,2) \to c$$

$$\nwarrow\ (1,2) \to (n_3, n_4)\ ,\ n_4 \to c$$

$$\nwarrow^{*}_{n_3=1, n_4=2, c=2}\ \square \qquad\qquad t\ (1,2) \to 2$$

# Esempio

$$t \triangleq \underbrace{\mathbf{rec}\ \underbrace{x}_{\tau}.\ \underbrace{x}_{\tau}}_{\tau}\ : \tau$$

$$\mathbf{rec}\ x.\ x \to c \ \nwarrow\ x[^{\mathbf{rec}\ x.\ x}/_x] \to c$$

$$= \mathbf{rec}\ x.\ x \to c$$

stesso goal da cui siamo partiti
nessun'altra opzione da esplorare:
divergenza!

# Esempio

$$fact \triangleq \mathbf{rec}\ f.\ \lambda x.\ \mathbf{if}\ x\ \mathbf{then}\ 1\ \mathbf{else}\ x \times (f\ (x-1))$$

$$fact \to c \ \nwarrow \ (\lambda x.\ \mathbf{if}\ x\ \mathbf{then}\ 1\ \mathbf{else}\ x \times (f(x-1)))[^{fact}/_{f}] \to c$$

$$= \lambda x.\ \mathbf{if}\ x\ \mathbf{then}\ 1\ \mathbf{else}\ x \times (\overbrace{(\mathbf{rec} f.\ ...)}^{fact}(x-1)) \to c$$

$$\nwarrow_{c = \lambda x.\ \mathbf{if}\ x\ \mathbf{then}\ 1\ \mathbf{else}\ x \times (fact(x-1))} \ \square$$

# Esempio

$$\mathit{fact} \triangleq \mathbf{rec}\ f.\ \lambda x.\ \mathbf{if}\ x\ \mathbf{then}\ 1\ \mathbf{else}\ x \times (f\ (x-1))$$

$$(\mathit{fact}\ 1) \to c \ \nwarrow\ \mathit{fact} \to \lambda x'.\ t'\ ,\ t'[^1/_{x'}] \to c$$

$$\nwarrow^*_{x'=x,t'=\mathbf{if}\dots}\ (\mathbf{if}\ x\ \mathbf{then}\ 1\ \mathbf{else}\ x \times (\mathit{fact}\ (x-1)))[^1/_x] \to c$$

$$= \mathbf{if}\ 1\ \mathbf{then}\ 1\ \mathbf{else}\ 1 \times (\mathit{fact}\ (1-1)) \to c$$

$$\nwarrow\ 1 \to n\ ,\ n \neq 0\ ,\ 1 \times (\mathit{fact}\ (1-1)) \to c$$

$$\nwarrow^*_{n=1,c=n_1 \underline{\times} n_2}\ 1 \to n_1\ ,\ (\mathit{fact}\ (1-1)) \to n_2$$

laziness
si vede qui

$$\nwarrow_{n_1=1}\ \mathit{fact} \to \lambda x''.\ t''\ ,\ t''[^{1-1}/_{x''}] \to n_2$$

$$\nwarrow^*_{x''=x,t''=\mathbf{if}\dots}\ (\mathbf{if}\ x\ \mathbf{then}\ 1\ \mathbf{else}\ x \times (\mathit{fact}\ (x-1)))[^{1-1}/_x] \to n_2$$

$$= \mathbf{if}\ 1-1\ \mathbf{then}\ 1\ \mathbf{else}\ (1-1) \times (\mathit{fact}\ ((1-1)-1)) \to n_2$$

$$\nwarrow\ 1-1 \to 0\ ,\ 1 \to n_2$$

$$\nwarrow^*_{n_2=1}\ \square \qquad c = n_1 \underline{\times} n_2 = 1 \underline{\times} 1 = 1$$

# HOFL
# Semantica operazionale Eager

# Lazy vs Eager

$$\frac{t_1 \to \lambda x.\, t_1' \quad t_1'[{}^{t_0}/_{x}] \to c}{(t_1\ t_0) \to c} \quad \text{(lazy)}$$

$$\frac{t_1 \to \lambda x.\, t_1' \quad t_0 \to c_0 \quad t_1'[{}^{c_0}/_{x}] \to c}{(t_1\ t_0) \to c} \quad \text{(eager)}$$

# Lazy vs Eager

$$t \triangleq (\lambda x.\ 1)\ (\mathbf{rec}\ y.\ y) : int$$

$x : \tau$
$y : \tau$

---

lazy

$$t \to c \ \nwarrow\ \lambda x.\ 1 \to \lambda x'.\ t'\ ,\ t'[^{\mathbf{rec}\ y.\ y}/_{x'}] \to c$$

$$\nwarrow_{x'=x,\ t'=1}\ 1[^{\mathbf{rec}\ y.\ y}/_{x}] \to c$$

$$= 1 \to c$$

$$\nwarrow_{c=1}\ \square$$

---

eager

$$t \to c \ \nwarrow\ \lambda x.\ 1 \to \lambda x'.\ t'\ ,\ \mathbf{rec}\ y.\ y \to c'\ ,\ t'[^{c'}/_{x'}] \to c$$

$$\nwarrow_{x'=x,\ t'=1}\ \mathbf{rec}\ y.\ y \to c'\ ,\ 1[^{c'}/_{x}] \to c$$

$$\nwarrow\ \mathbf{rec}\ y.\ y \to c'\ ,\ 1[^{c'}/_{x}] \to c$$

divergenza!

# Lazy vs Eager

$$t \triangleq (\lambda x.\ x + x)\ (1 \times 2)\ : int \qquad\qquad x : int$$

---

$$t \to c \ \nwarrow\ \lambda x.\ x + x \to \lambda x'.\ t'\ ,\ t'[^{1\times 2}/_{x'}] \to c$$

lazy $\nwarrow_{x'=x,\ t'=x+x}\ (x + x)[^{1\times 2}/_x] \to c$

$$= (1 \times 2) + (1 \times 2) \to c$$

valutata due volte

$$\nwarrow_{c=c_1 \underline{+} c_2}\ \boxed{(1 \times 2) \to c_1\ ,\ (1 \times 2) \to c_2}$$

$$\nwarrow^{*}_{c_1=2, c_2=2}\ \square \qquad c = c_1 \underline{+} c_2 = 2 \underline{+} 2 = 4$$

---

$$t \to c \ \nwarrow\ \lambda x.\ x + x \to \lambda x'.\ t'\ ,\ 1 \times 2 \to c'\ ,\ t'[^{c'}/_{x'}] \to c$$

eager $\nwarrow_{x'=x,\ t'=x+x}\ 1 \times 2 \to c'\ ,\ (x + x)[^{c'}/_x] \to c$

$$\nwarrow^{*}_{c'=2}\ (x + x)[^{2}/_x] \to c$$

$$= 2 + 2 \to c$$

$$\nwarrow^{*}_{c=4}\ \square$$

# HOFL
# Proprieta' della semantica operazionale

# Terminazione

termina? $\forall t.\ \exists c.\ t \to c?$ ✖ $\mathbf{rec}\ x.\ x$

# Determinismo?

determinismo? $\forall t.\ \forall c_1, c_2.\ t \to c_1 \wedge t \to c_2 \Rightarrow c_1 = c_2\,?$ ✓

$$P(t \to c) \triangleq \forall c_1.\ t \to c_1 \Rightarrow c_1 = c$$

per ind. strutturale (provateci!)

# Subject reduction

i tipi assegnati staticamente non cambiano in fase di esecuzione

subject reduction? $\forall t.\ \forall c.\ \forall \tau.\ t \to c \wedge t : \tau \Rightarrow c : \tau\ ?$ ✓

$$P(t \to c) \triangleq \forall \tau.\ t : \tau \Rightarrow c : \tau$$

per ind. strutturale (provateci!)

# Congruenza?

$$t_1 \equiv_{\mathrm{op}} t_2 \quad \text{iff} \quad \forall c.\ (t_1 \to c \Leftrightarrow t_2 \to c)$$

e' una congruenza? ❌

$$2 \equiv_{\mathrm{op}} 1 + 1$$

$$\lambda x.\ 2 \not\equiv_{\mathrm{op}} \lambda x.\ 1 + 1$$

$$\lambda x.\ 2\ ,\ \ \lambda x.\ 1 + 1 \in C_{\tau \to int}$$

$$\lambda x.\ 2 \to \lambda x.\ 2$$

$$\lambda x.\ 1 + 1 \to \lambda x.\ 1 + 1$$