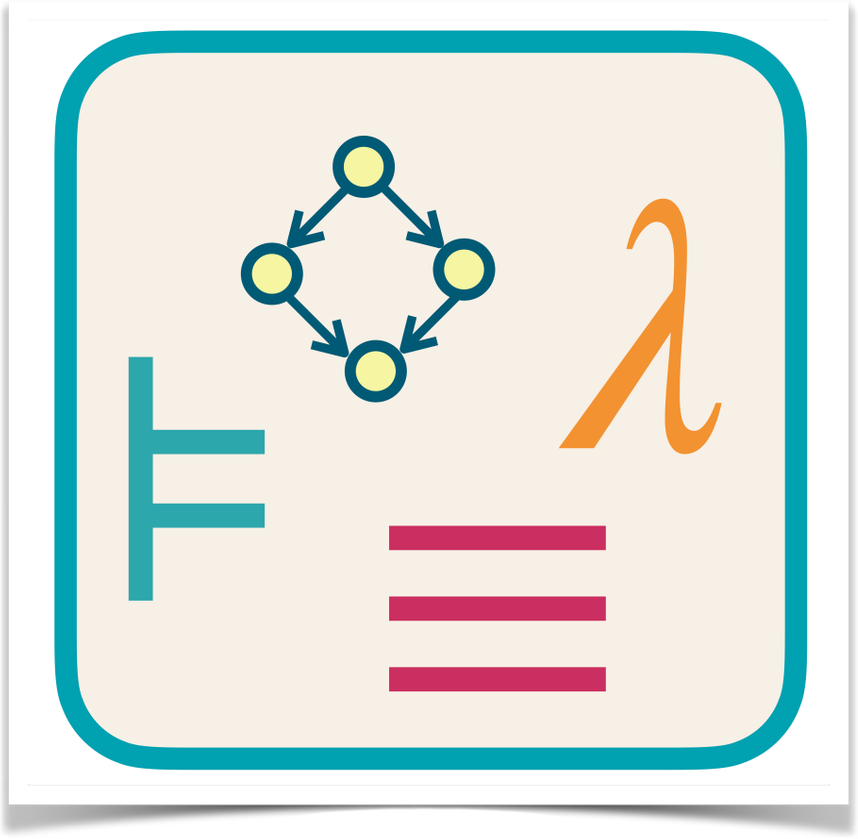


LPP 2026/27 (063AA, 9CFU)

Programming Languages

Fabio Gadducci

and Laboratory

Vincenzo Ciancia

11b - HOFL Operational Semantics

Disclaim

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

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

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

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

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

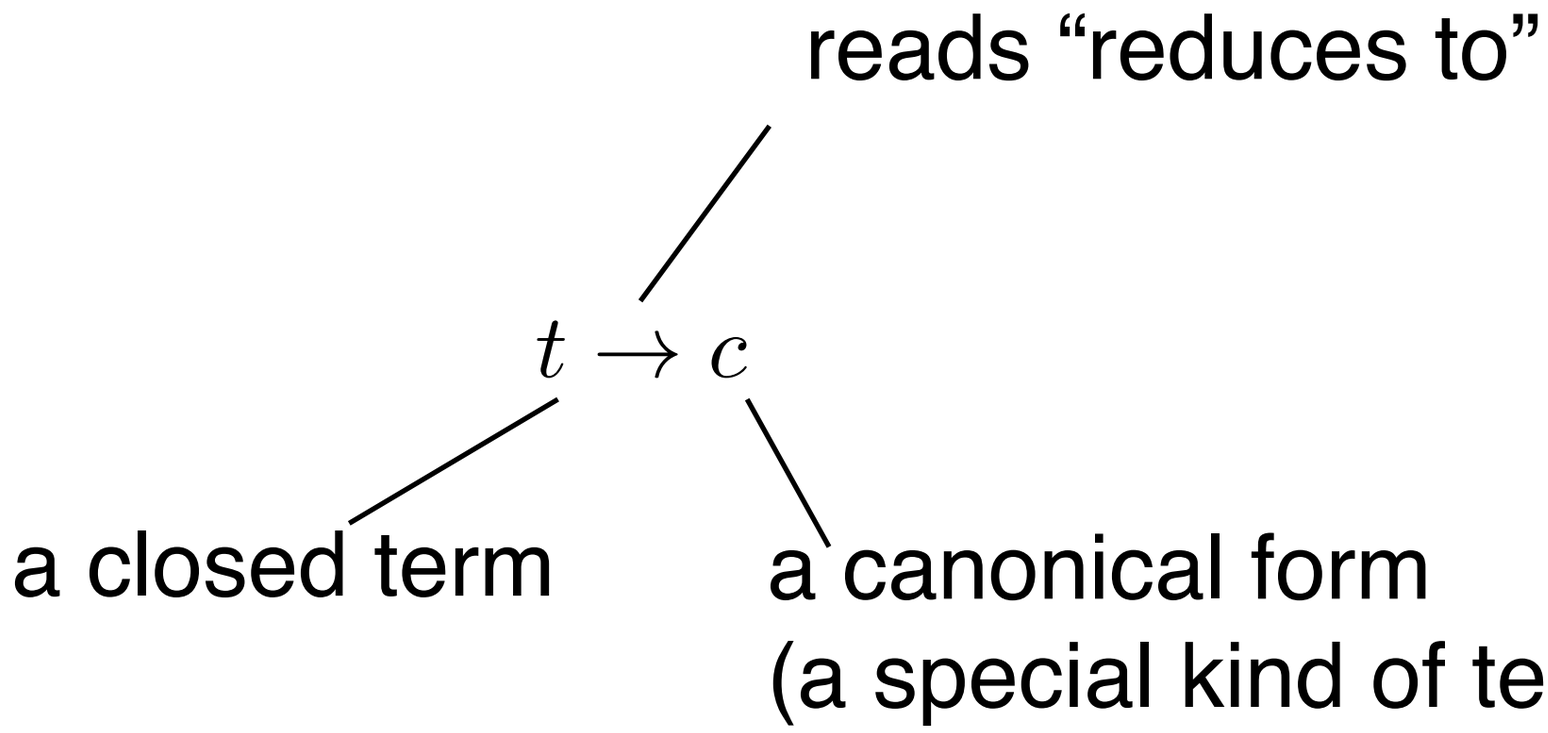
$$
\frac{x : \tau \quad t : \tau}{\textbf{rec } x.\ t : \tau}
$$

we assign semantics only to terms that are: well-formed and closed

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

Canonical forms

Statements



Big step operational semantics

computation of canonical form
(by term manipulation)

Canonical forms

set of canonical forms $C_\tau \subseteq T_\tau$
with type τ

(laziness)
not required to be
in canonical forms

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

t not necessarily
a closed term

$$\frac{\lambda x.\ t : \tau_0 \to \tau_1 \quad \lambda x.\ t \text{ closed}}{\lambda x.\ t \in C_{\tau_0 \to \tau_1}}$$

Canonical forms?

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

$1 + 2 \times 3$ ✗ **if** 0 **then** 0 **else** 0 ✗

$(1, 2)$ ✓ $\lambda x.\ 1$ ✓

$(1 + 2, 2 - 1)$ ✓ $\lambda x.\ 1 + 2 \times 3$ ✓

$\mathbf{fst}(1, 2)$ ✗ $\lambda x.\ \mathbf{fst}(1, 2)$ ✓

HOFL

Lazy operational semantics

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

Lazy op semantics

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

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

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

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

Type system (remind)

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

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

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

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

Example

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

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

Example

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

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

laziness:
no need to evaluate 1+2

Example

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

Example (ctd)

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

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

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

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

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

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

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

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

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

Example

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

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

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

same goal from which we started
no other option to explore:
divergence!

Example

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

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

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

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

Example

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

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

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

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

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

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

laziness evident here

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

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

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

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

$$\nwarrow^{*}_{n_2=1}\ \square$$

$$c = n_1 \underline{\times} n_2 = 1 \underline{\times} 1 = 1$$

HOFL

Eager operational semantics

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

$$t \triangleq (\lambda x.\ 1)\ (\mathbf{rec}\ y.\ y)\ :\ int \qquad \begin{array}{r} x : \tau \\ y : \tau \end{array}$$

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

lazy

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

$$= 1 \to c$$

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

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

eager

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

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

divergence!

Lazy vs Eager

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

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

lazy

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

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

evaluated twice

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

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

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

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

Properties of operational semantics

Termination

termination? $\forall t.\ \exists c.\ t \to c?$ ✗ $\mathbf{rec}\ x.\ x$

Determinacy?

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

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

by rule induction (try by yourself)

Subject reduction

(statically assigned types do not change at runtime)

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

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

by rule induction (try by yourself)

Congruence?

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

is it a congruence? ✗

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

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

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

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

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

HOFL, type inference and operational semantics

[**Ex. 1**] Determine the type of the HOFL term

$$t \stackrel{\text{def}}{=} \mathbf{rec}\ x.\ ((\lambda y.\ \mathbf{if}\ y\ \mathbf{then}\ 0\ \mathbf{else}\ 0)\ x).$$

Then compute its (lazy) canonical form.

Ex. 1, typing

$$t \triangleq \mathbf{rec}\ \underbrace{x}_{int}.\ (\ (\ \underbrace{\lambda y}_{int}.\ \underbrace{\mathbf{if}\ \underbrace{y}_{int}\ \mathbf{then}\ \underbrace{0}_{int}\ \mathbf{else}\ \underbrace{0}_{int}}_{int}\ )\ \underbrace{x}_{int}\ ) \quad : int$$

$int \to int$

int

int

Ex. 1, canonical form?

$$t \triangleq \mathbf{rec}\ x.\ (\ (\ \lambda y.\ \mathbf{if}\ y\ \mathbf{then}\ 0\ \mathbf{else}\ 0\ )\ x\ ) \quad : int$$

$t \to c \nwarrow\ (\ (\ \lambda y.\ \mathbf{if}\ y\ \mathbf{then}\ 0\ \mathbf{else}\ 0\ )\ x\ )[^{t}/_{x}] \to c$

$= (\ (\ \lambda y.\ \mathbf{if}\ y\ \mathbf{then}\ 0\ \mathbf{else}\ 0\ )\ t\ ) \to c$

$\nwarrow\ \lambda y.\ \mathbf{if}\ y\ \mathbf{then}\ 0\ \mathbf{else}\ 0 \to \lambda x'.\ t'\ ,\ t'[^{t}/_{x'}] \to c$

$\nwarrow_{x'=y, t'=\mathbf{if}\ldots}\ (\ \mathbf{if}\ y\ \mathbf{then}\ 0\ \mathbf{else}\ 0\ )[^{t}/_{y}] \to c$

$= (\ \mathbf{if}\ t\ \mathbf{then}\ 0\ \mathbf{else}\ 0\ ) \to c$

$\nwarrow\ t \to n\ ,\ 0 \to c$ (it doesn't matter if $n = 0$)

same goal from which we started:
no canonical form

[**Ex. 2**] Determine the type of the HOFL term

$$map \stackrel{\text{def}}{=} \lambda f.\ \lambda x.\ ((f\ \mathbf{fst}(x)), (f\ \mathbf{snd}(x)))$$

Then, compute the (lazy) canonical forms of the terms

$$t_1 \stackrel{\text{def}}{=} map\ (\lambda z.\ 2 \times z)\ (1, 2) \qquad t_2 \stackrel{\text{def}}{=} \mathbf{fst}\ (\ map\ (\lambda z.\ 2 \times z)\ (1, 2)\ )$$

Ex. 2, typing

$$map \triangleq \lambda \underbrace{f}_{\tau_1 \to \tau} . \lambda \underbrace{x}_{\tau_1 * \tau_1} . \Big(\big(\underbrace{f}_{\tau_1 \to \tau} \underbrace{\mathbf{fst}(x)}_{\tau_1 * \tau_2} \big)_{\tau_1}, \big(\underbrace{f}_{\tau_1 \to \tau} \underbrace{\mathbf{snd}(x)}_{\tau_1 * \tau_2} \big)_{\tau_2 = \tau_1} \Big)_{\tau * \tau}$$

$$\tau_1 * \tau_1 \to \tau * \tau$$

$$(\tau_1 \to \tau) \to \tau_1 * \tau_1 \to \tau * \tau$$

$$map \ : \ (\tau_1 \to \tau) \to \tau_1 * \tau_1 \to \tau * \tau$$

Ex. 2a, canonical form

$$map \triangleq \lambda f\ .\ \lambda x\ .\ (\ (\ f\ \mathbf{fst}(x)\ )\ ,\ (\ f\ \mathbf{snd}(x)\ )\ )$$

$$t_1 \triangleq map\ (\lambda z.\ 2 \times z)\ (1,2)$$

$$t_1 \to c \nwarrow (map\ (\lambda z.\ 2 \times z)) \to \lambda x'.\ t'\ ,\ t'[^{(1,2)}/_{x'}] \to c$$

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

$$\nwarrow_{f'=f,t''=\lambda x\ldots} (\lambda x.\ ((f\ \mathbf{fst}(x)),(f\ \mathbf{snd}(x))))[^{\lambda z.\ 2\times z}/_{f}] \to \lambda x'.\ t'\ ,\ t'[^{(1,2)}/_{x'}] \to c$$

$$= (\lambda x.\ (((\lambda z.\ 2 \times z)\ \mathbf{fst}(x)),((\lambda z.\ 2 \times z)\ \mathbf{snd}(x)))) \to \lambda x'.\ t'\ ,\ t'[^{(1,2)}/_{x'}] \to c$$

$$\nwarrow_{x'=x,t'=(\ldots,\ldots)} (((\lambda z.\ 2 \times z)\ \mathbf{fst}(x)),((\lambda z.\ 2 \times z)\ \mathbf{snd}(x)))[^{(1,2)}/_{x}] \to c$$

$$= (((\lambda z.\ 2 \times z)\ \mathbf{fst}(1,2)),((\lambda z.\ 2 \times z)\ \mathbf{snd}(1,2))) \to c$$

$$\nwarrow_{c=(((\lambda z.\ 2\times z)\ \mathbf{fst}(1,2)),((\lambda z.\ 2\times z)\ \mathbf{snd}(1,2)))} \square$$

Ex. 2b, canonical form

$$t_1 \to (\ ((\lambda z.\ 2 \times z)\ \mathbf{fst}(1,2))\ ,\ ((\lambda z.\ 2 \times z)\ \mathbf{snd}(1,2))\ )$$

$$\mathbf{fst}(t_1) \to c \ \nwarrow\ t_1 \to (t_1', t_2')\ ,\ t_1' \to c$$

$$\nwarrow^{*}_{t_1'=(\lambda z.\ 2\times z)\ \mathbf{fst}(1,2)\ ,\ t_2'=(\lambda z.\ 2\times z)\ \mathbf{snd}(1,2)} (\lambda z.\ 2 \times z)\ \mathbf{fst}(1,2) \to c$$

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

$$\nwarrow_{z'=z,t'=2\times z}\ (2 \times z)[{}^{\mathbf{fst}(1,2)}/_{z}] \to c$$

$$= (2 \times \mathbf{fst}(1,2)) \to c$$

$$\nwarrow_{c=n_1 \underline{\times} n_2}\ 2 \to n_1\ ,\ \mathbf{fst}(1,2) \to n_2$$

$$\nwarrow^{*}_{n_1=2}\ (1,2) \to (t_1'', t_2'')\ ,\ t_1'' \to n_2$$

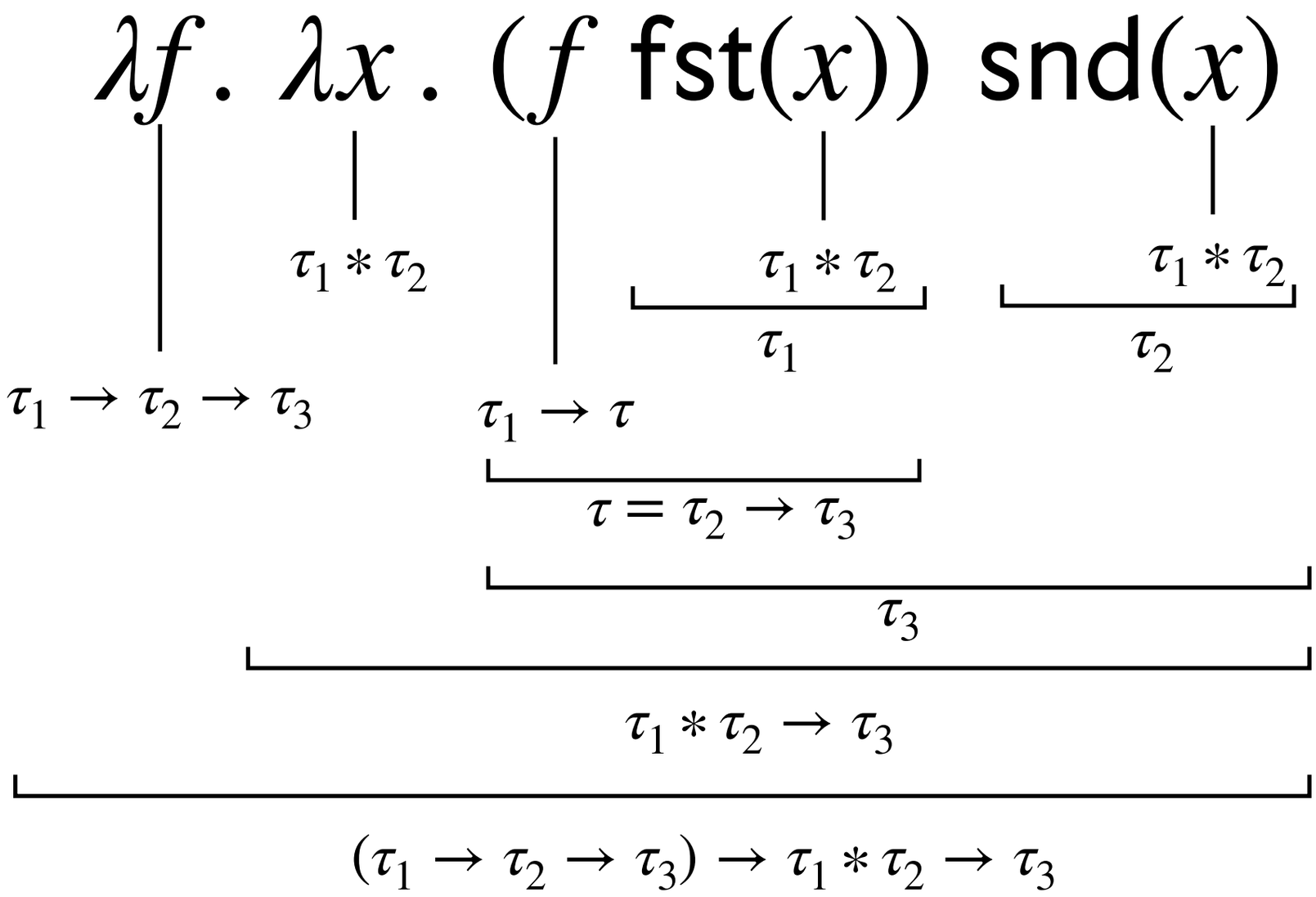
$$\nwarrow_{t_1''=1,t_2''=2}\ 1 \to n_2$$

$$\nwarrow_{n_2=1}\ \square$$

$$c = n_1 \underline{\times} n_2 = 2 \underline{\times} 1 = 2$$

Ex. 3, part 1

Determine the type of the HOFL term



Which function is it? *uncurry*

Ex. 3, part 2

$$uncurry \triangleq \lambda f.\ \lambda x.\ (f\ \mathsf{fst}(x))\ \mathsf{snd}(x)$$

Prove that for every $t : \tau_1 \to \tau_2 \to \tau_3$ and $t_0 : \tau_1 * \tau_2$ then

$$(t\ \mathsf{fst}(t_0))\ \mathsf{snd}(t_0) \equiv_{\mathrm{op}} (uncurry\ t)\ t_0$$

$$(uncurry\ t)\ t_0 \to c \nwarrow (uncurry\ t) \to \lambda x'.\ t', \quad t'[{}^{t_0}/_{x'}] \to c$$

$$\nwarrow uncurry \to \lambda f'.\ t'', \quad t''[{}^{t}/_{f'}] \to \lambda x'.\ t', \quad t'[{}^{t_0}/_{x'}] \to c$$

$$\nwarrow (\lambda x.\ (f\ \mathsf{fst}(x))\ \mathsf{snd}(x))[{}^{t}/_{f}] \to \lambda x'.\ t', \quad t'[{}^{t_0}/_{x'}] \to c$$

$$= (\lambda x.\ (t\ \mathsf{fst}(x))\ \mathsf{snd}(x)) \to \lambda x'.\ t', \quad t'[{}^{t_0}/_{x'}] \to c$$

$$\nwarrow ((t\ \mathsf{fst}(x))\ \mathsf{snd}(x))[{}^{t_0}/_{x}] \to c$$

$$= (t\ \mathsf{fst}(t_0))\ \mathsf{snd}(t_0) \to c$$