

Simply typed λ calculus

if true then 0 else false

X type systems are conservative

$$T ::= \text{Bool} \mid T \rightarrow T$$

E.g. $\text{Bool} \rightarrow \text{Bool}$

$\text{Bool} \rightarrow (\text{Bool} \rightarrow \text{Bool})$

$\text{int} \not\models (\text{int } x)$

Inhibition

arguments have types

Typed λ calc

$t = x$

$\mid \lambda x:T. t$

$\mid t \ t$

a value is either
a Boolean or $(\lambda x \dots)$

$(\lambda x:T. t) \ v \rightarrow$
 $[x \mapsto v]t$

$t_1 \rightarrow t'_1$

$t_1 \ t_2 \rightarrow t'_1 \ t_2$

$$\frac{t \in \text{Nat}}{\text{isZero } t \in \text{Bool}} T_{\text{isZero}} \quad \left(\text{Recall} \right)$$

$$\frac{t_2 \longrightarrow t'_2}{v \ t_2 \longrightarrow v \ t'_2}$$

typing context

$$\frac{\boxed{x \in T_1} \vdash t \in T_2}{\lambda x : T_1. t \in T_1 \rightarrow T_2} T_{\text{ABS}}$$

if x has type T_1 then I can prove $t \in T_2$

Induction

$$x \in \text{Int} \vdash x + x \in \text{Int}$$

a typing context Γ is a sequence of
the form $\alpha_1 \in T_1, \dots, \alpha_n \in T_n$

Γ can be \emptyset (empty)

$$\underbrace{\Gamma, \alpha_1 \in T_1}$$

T_{ABS}

α is
no longer
free

$$\frac{\Gamma, \alpha \in T_1 \vdash t \in T_2}{\Gamma \vdash \lambda x:T_1. t \in T_1 \rightarrow T_2}$$

T_{APP}

$$\frac{\Gamma \vdash t_1 \in T_1 \rightarrow T \quad \Gamma \vdash t_2 \in T_1}{\Gamma \vdash t_1 \ t_2 : T}$$

T_{var}

$$\frac{x:T \in \Gamma}{\Gamma \vdash x \in T}$$

the typing context is a set

T_{if}

$$\frac{\Gamma \vdash t_1 \in \text{Bool} \quad \Gamma \vdash t_2 \in T \quad \Gamma \vdash t_3 \in T}{\Gamma \vdash \text{if } t_1 \text{ then } t_2 \text{ else } t_3 \in T}$$

if true then $(\lambda x:\text{Bool}. x)$ else $(\lambda x:\text{Bool}. \text{not } x)$
 $\in (\text{Bool} \rightarrow \text{Bool})$

$x \in \text{Bool} \equiv x : \text{Bool}$

✓ typing context

$\frac{x : \text{Bool} \in \{x \in \text{Bool}\}}{T_{\text{VAR}}}$

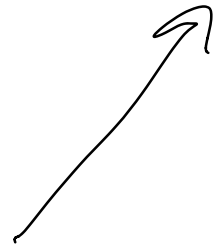
$\frac{x : \text{Bool} \vdash x \in \text{Bool}}{T_{\text{ABS}}}$

$\phi \vdash \lambda x : \text{Bool}. x \in \text{Bool} \rightarrow \text{Bool}$

✓
 $\phi \vdash \text{true} \in \text{Bool}$

$\phi \vdash (\lambda x : \text{Bool}. x) \text{ true} \in \text{Bool}$

T_{APP}



Inversion lemma (Just reading rules upwards)

If $\Gamma \vdash x \in T$ then $x \in T \in \Gamma$

If $\Gamma \vdash \lambda x:T. t \in R$ then

$R = T \rightarrow R_2$ for

some R_2

and $\Gamma, x:T \vdash t \in R_2$

\vdots

Uniqueness theorem

In a given typing context Γ , a term t ,
with free variables in Γ , has at most
one type

$$\frac{\Gamma}{\Gamma \vdash x \in \text{Bool} \vdash x \in \text{Bool}}$$
$$\frac{\Gamma}{\Gamma \vdash x \in \text{Bool} \rightarrow \text{Bool} \vdash x \in \text{Bool} \rightarrow \text{Bool}}$$

Canonical forms lemma

① if v has type Bool , then v is either `true`/`false`

② if v has type $T_1 \rightarrow T_2$ then

$$v = \lambda x:T_1. t$$

PROGRESS THM

Suppose $\emptyset \vdash t \in T$

Then t is a value or $t \rightarrow t'$

PROOF Induction on typing derivation

the only interesting case T_{App}

imagine t is of form $t_1 t_2$

By hypothesis t_1 is a value or $t_1 \rightarrow t'_1$

case \rightarrow - if $t_1 \rightarrow t'_1$ then $t \rightarrow t'_1 t_2$

case V - if t_1 is a value

case B1

t_2 can take a step
 $t \rightarrow t_1 t_2'$

case B2

t_1 and t_2 are values

By canonical forms

$$t_1 = (\lambda x. t_1')$$

this means

$$t \longrightarrow t_1'$$

$$[x \mapsto t_2] t_1'$$

PRESERVATION

If $\Gamma \vdash t \in T$ and

$$t \longrightarrow t'$$

then

$$\Gamma \vdash t' \in T$$

Erase theorem

$$\text{erase}(x) = x$$

$$\text{erase}(\lambda x:T. t) = \lambda x. t$$

$$\text{erase}(t_1 \ t_2) = \text{erase}(t_1) \ \text{erase}(t_2)$$

Then

- If $t \rightarrow t'$ under typed evaluation
then $\text{erase}(t) \rightarrow \text{erase}(t')$

- If $\text{erase}(t) \rightarrow m'$ then
there is a typed term t' s.t.
 $t \rightarrow t'$ and $\text{erase}(t') = m'$