Simply-typed λ-calculus

$$\omega \equiv (\lambda x.\, x x) \qquad \Omega = \omega\omega$$

Types $\quad A, B ::= \iota \mid A \to B \quad [\, A \times B \mid 1 \,]$

Terms $\quad M, N ::= x \mid (\lambda x{:}A.\, M) \mid M\, N$

$\qquad \mid \langle M, N\rangle \mid \pi_1\, M \mid \pi_2\, N \mid (\ast)$

Values $::=$ $\lambda x{:}A.\, M$

$\langle M, N\rangle \leftarrow$ cbn

$\langle V, W\rangle \leftarrow$ cbv

$(\ast) \leftarrow$ cbn/cbv

"Stuck" $\quad \pi_1\, (\lambda x{:}A.\, x) \not\to$

$(\ast)\, \langle (\ast), (\ast)\rangle \not\to$

$$(\lambda x{:}\iota.\, x) : \iota \to \iota$$

$$\boxed{\begin{array}{lll} \pi_1\, \langle M, N\rangle & \to_{\substack{bn\\ \beta}} & M \\ \pi_2\, \langle M, N\rangle & \to_{\substack{bn\\ \beta}} & N \end{array}} \qquad \pi_1\, \langle V, W\rangle \to V$$

The type system

Contexts $\Gamma ::= \cdot \mid \Gamma, x : A$

Γ_1, Γ_2 to "append" Γ_1, Γ_2

$\mathrm{dom}(\Gamma) = \{ x \mid x : A \text{ appears in } \Gamma \}$

$(x : A) \in \Gamma$ to mean $x : A$ appears in Γ

$\mathrm{dom}(\Gamma_1) \cap \mathrm{dom}(\Gamma_2) = \emptyset$

Typing Relation : $\boxed{\Gamma \vdash M : A}$ judgment

(var) $$\frac{}{\Gamma \vdash x : A}\ (x : A \in \Gamma)$$

(abs) $$\frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x : A. M : A \to B}\ (x \notin \mathrm{dom}(\Gamma))$$

(app) $$\frac{\Gamma \vdash M : A \to B \qquad \Gamma \vdash N : A}{\Gamma \vdash M\,N : B}$$

Q: does there exist a typing derivation for $\underline{\Gamma} \vdash (\lambda x : \underline{A} . x\, x) : \underline{B}$?

Ans: No? there is no type T such that $T = T \to T$.

$$\frac{\Gamma \vdash M : A \qquad \Gamma \vdash N : B}{\Gamma \vdash \langle M, N \rangle : A \times B}$$

$$\frac{\Gamma \vdash M : A \times B}{\Gamma \vdash \underline{\pi_1 M} : A} \qquad \frac{\Gamma \vdash M : A \times B}{\Gamma \vdash \pi_2 M : B}$$

$$\Gamma \vdash \langle * \rangle : 1$$

$$\frac{\overline{\Gamma \vdash (\lambda x{:}A.\, x) : B \times C}}{\Gamma \vdash \underline{\pi_1 (\lambda x{:}A.\, x)} : B}$$ ✗ "wrong" or "stuck"

<u>Type Safety</u>: If $\cdot \vdash M : A$ and $M \twoheadrightarrow_{\substack{bn\\bv\\n}} M'$ then $\cdot \vdash M' : A$.

Milner's "well-typed programs don't go wrong"

Wright & Felleisen 1994 Syntactic Type Safety

- (β) $\quad (\lambda x:A.M)\ N \to_\beta M[N/x]$

  (ξ) $\quad \dfrac{M \to_\beta M'}{\lambda x:A.M \to_\beta \lambda x:A.M'}$

Lemma Subject Reduction

If $\Gamma \vdash M : A$ and $M \to_\beta M'$
then $\Gamma \vdash M' : A$.

Proof

By induction on the structure of M.

- case $M = x$ then $x \not\to_\beta$ ✓

- case $M = \lambda x:B.M'$

  ▫ $\dfrac{\Gamma, x:B \vdash M' : C}{\Gamma \vdash \lambda x:B.\ M' : \underbrace{B \to C}_{=A}}$ $\quad$ ▫ $M' \to M''$ by (ξ)

  $\Gamma, x:B \vdash M'' : C$ (by I.H.)

  by abs $\quad \Gamma \vdash \lambda x:B.M'' : \underbrace{B \to C}_{=A}$ ✓

- Case $(M_1 M_2)$ $\quad$ a) β $\quad M_1 = \lambda x:B.M_1'$
  $M' = M_1'[M_2/x]$

  b) $M_1 \to M_1'$
  $M' = M_1' M_2$ $\quad$ c) cong_2

Lemma **Substitution**

If $\Gamma_1, x:A, \Gamma_2 \vdash M : B$ and

$\Gamma_1 \vdash N : A$

then $\Gamma_1, \Gamma_2 \vdash M[N/x] : B$.

Proof: By induction on the structure of M.

- case $M = y$. $\Gamma_1, x:A, \Gamma_2 \vdash y : B$
  - $x = y$ $\quad \dfrac{}{\Gamma_1, x:A, \Gamma_2 \vdash x : B}\,(x:A \in \ldots)$

    $A = B$

    $x[N/x] = N$

    to show: $\Gamma_1, \Gamma_2 \vdash N : A = B$ ✓

Lemma **Weakening** if $\Gamma_1 \vdash M : A$

and Γ_1, Γ_2 is well formed

then $\Gamma_1, \Gamma_2 \vdash M : A$

Proof

case $M = x$ $\quad \dfrac{}{\Gamma_1 \vdash x : A}\,(x:A \in \Gamma_1) \qquad \dfrac{}{\Gamma_1, \Gamma_2 \vdash x : A}\,(x:A \in \Gamma_1, \Gamma_2)$

Rest by

- case $x \neq y$ $\quad M[N/x] = y[N/x] = y$

$\Gamma_1, x : A, \Gamma_2 \vdash y : B$ $\quad \boxed{\Gamma_1, \Gamma_2 \vdash y : B}$ to show

Lemma
Strengthening If $\Gamma_1, x : A, \Gamma_2 \vdash M : B$
and $x \notin fv(M)$
then $\Gamma_1, \Gamma_2 \vdash M : B$

Proof By induction on M.

- Case $M = \lambda y : C . M'$

have $\Gamma_1, x : A, \Gamma_2 \vdash \lambda y : C . M' : C \to D$

by inversion $\Gamma_1, x : A, \Gamma_2, y : C \vdash M' : D$
and $x \neq y$.

$$(\lambda y : C . M')[N/x] = \lambda y : C . (M'[N/x])$$

By I.H. with $\Gamma_2 := \Gamma_2, y : C$ and $B := D$

we have $\Gamma_1 \quad \Gamma_2, y : C \vdash M'[N/x] : D$

use abs. $\Gamma_1 \quad \Gamma_2 \vdash \lambda y : C . (M'[N/x]) : \underbrace{C \to D}_{B}$ ✓

- Case $M = M_1\ M_2$

by I.H. twice, we have

$$\frac{\Gamma_1, \Gamma_2 \vdash M_1[N/x] : C \to B \qquad \Gamma_1, \Gamma_2 \vdash M_2[N/x] : C}{\Gamma_1, \Gamma_2 \vdash (M_1\ M_2)[N/x] : B} \checkmark$$

- Preservation: $\cdot \vdash M : A$ and $M \to_\beta M'$
  then $\cdot \vdash M' : A$

- Progress: If $\cdot \vdash M : A$
  then either M is a value ($\lambda x{:}A.\ M$)
  or there exists M' s.t.
  $M \to M'$.

Corollary: if $\vdash M : A$ then M is not stuck.

$\boxed{\pi_1\ (\lambda x.x)} \not\to$