Ejercicio 1 ★

Determinar qué expresiones son sintácticamente válidas (es decir, pueden ser generadas con las gramáticas presentadas) y determinar a qué categoría pertenecen (expresiones de términos o expresiones de tipos):

a) x ✓

b) $x\ x$ ✓

c) M No es una expresión

d) $M\ M$ //

e) true false ✓

f) true succ(false true) ✓

g) λx.isZero(x) No, falta el tipo

h) $\lambda x : \sigma$. succ(x) Falta el tipo

i) λx: Bool. succ(x) ✓

j) λx: if true then Bool else Nat. x ? No

k) σ No

l) Bool ✓

m) Bool $\to$ Bool ✓

n) Bool $\to$ Bool $\to$ Nat ✓

ñ) (Bool $\to$ Bool) $\to$ Nat ✓

de Tipo

El resto Términos

o) succ true ✓

p) λx: Bool. if zero then true else zero succ(true) ✓

Ejercicio 2

Mostrar un término que utilice al menos una vez **todas** las reglas de generación de la gramática de los términos y exhibir su *árbol sintáctico*.

(λx: Bool. if x then True False else isZero(pred(succ(zero)))) y

Ejercicio 3 ★

a) Marcar las ocurrencias del término x como subtérmino en λx: Nat. succ((λx: Nat. x) x).

b) ¿Ocurre x_1 como subtérmino en λx_1: Nat. succ(x_2)?

c) ¿Ocurre $x\ (y\ z)$ como subtérmino en $u\ x\ (y\ z)$?

a) λx: Nat. succ((λx: Nat. x) x).

b) x, no ocurre como subtérmino

c) u x (y z)

No ocurre

Ejercicio 4 ★

Para los siguientes términos:

a) $u\ x\ (y\ z)\ (\lambda v$: Bool. $v\ y)$

b) $(\lambda x$: Bool $\to$ Nat $\to$ Bool. λy: Bool $\to$ Nat. λz: Bool. $x\ z\ (y\ z))\ u\ v\ w$

c) $w\ (\lambda x$: Bool $\to$ Nat $\to$ Bool. λy: Bool $\to$ Nat. λz: Bool. $x\ z\ (y\ z))\ u\ v$

Se pide:

I Insertar todos los paréntesis de acuerdo a la convención usual.

II Dibujar el árbol sintáctico de cada una de las expresiones.

III Indicar en el árbol cuáles ocurrencias de variables aparecen ligadas y cuáles libres.

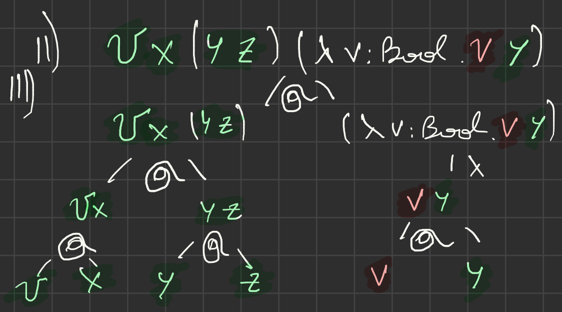
IV ¿En cuál o cuáles de los términos anteriores ocurre la siguiente expresión como subtérmino?
$(\lambda x$: Bool $\to$ Nat $\to$ Bool. λy: Bool $\to$ Nat. λz: Bool. $x\ z\ (y\ z))\ u$

a) I) u x (y z) (λv: bool . v y)

(((u x) (y z)) (λv: bool . v y))

II)

III)



b) $(\lambda x\colon \mathsf{Bool} \to \mathsf{Nat} \to \mathsf{Bool}.\ \lambda y\colon \mathsf{Bool} \to \mathsf{Nat}.\ \lambda z\colon \mathsf{Bool}.\ x\ z\ (y\ z))\ u\ v\ w$

$(((\lambda x : Bool \to Nat \to Bool.\ \lambda y : Bool \to Nat.\ \lambda z : Bool.\ (x\ z)(y\ z))\ u)\ v)\ w)$

II) $\lambda x : Bool \to Nat \to Bool.\ \lambda y : Bool \to Nat.\ \lambda z : Bool.\ x\ z\ (y\ z))\ u\ v\ w$

Acá ocurre el subtérmino $\#V$

$\lambda x : Bool \to Nat \to Bool.\ \lambda y : Bool \to Nat.\ \lambda z : Bool.\ x\ z\ (y\ z)\ u$ — v — w

λ

$\lambda y : Bool \to Nat.\ \lambda z : Bool.\ x\ z\ (y\ z))$

λ

$\lambda z : Bool.\ x\ z\ (y\ z)$

λ

$x\ z\ (y\ z)$

$x\ z$ — $y\ z$

x — z — y — z

c) $(w\ (\lambda x\colon \mathsf{Bool} \to \mathsf{Nat} \to \mathsf{Bool}.\ \lambda y\colon \mathsf{Bool} \to \mathsf{Nat}.\ \lambda z\colon \mathsf{Bool}\ (x\ z)(y\ z))\ u)\ v$

v

$w\ (\lambda x\colon \mathsf{Bool} \to \mathsf{Nat} \to \mathsf{Bool}.\ \lambda y\colon \mathsf{Bool} \to \mathsf{Nat}.\ \lambda z\colon \mathsf{Bool}\ (x\ z)(y\ z))$ — u

w — $\lambda x : Bool \to Nat \to Bool.\ \lambda y : Bool \to Nat.\ \lambda z : Bool\ (x\ z(y\ z))$

$\lambda\ \lambda\ \lambda$

$(x\ z\ (y\ z))$

$x\ z$ — $y\ z$

x — z — y — z

Ejercicio 5

Mostrar un término que no sea tipable y que no tenga variables libres ni abstracciones.

Zero True , True False

Ejercicio 6 *(Derivaciones ★)*

Dar una derivación –o explicar por qué no es posible dar una derivación– para cada uno de los siguientes juicios de tipado:

a) $\vdash$ if true then zero else succ(zero) : Nat

b) $x : \mathsf{Nat}, y : \mathsf{Bool} \vdash$ if true then false else $(\lambda z\colon \mathsf{Bool}.\ z)$ true : Bool

c) $\vdash$ if $\lambda x\colon \mathsf{Bool}.\ x$ then zero else succ(zero) : Nat

d) $x : \mathsf{Bool} \to \mathsf{Nat}, y : \mathsf{Bool} \vdash x\ y$: Nat

a)

$$\dfrac{\dfrac{}{\vdash True : Bool}\ T\text{-}True \qquad \dfrac{}{\vdash Zero : Nat}\ T\text{-}Zero \qquad \dfrac{\dfrac{}{\vdash Zero : Nat}\ T\text{-}zero}{\vdash 1 : Nat}\ T\text{-}succ}{\vdash if\ True\ then\ Zero\ else\ succ(Zero) : Nat}\ T\text{-}if$$

b)

$$\dfrac{\dfrac{}{\Gamma \vdash True : Bool}\ T \qquad \dfrac{}{\Gamma \vdash False\ Bool} \qquad \dfrac{\dfrac{\dfrac{}{\Gamma, z : Bool \vdash z : Bool}\ T\text{Var}}{\Gamma \vdash \lambda z : Bool.\ z : Bool \to Bool}\ ABS \qquad \dfrac{}{True : Bool}\ True}{\Gamma \vdash (\lambda z : Bool.\ z)\ true : Bool}\ APP}{\underbrace{x : Nat, y : Bool}_{\Gamma} \vdash if\ true\ then\ False\ else\ (\lambda z : Bool.\ z)\ true : Bool}\ if$$

c)

$\lambda x : Bool.\ x : Bool$ no es derivable

$$\dfrac{\vdash \lambda x : Bool.\ x : Bool \qquad \vdash Zero : Nat \qquad \vdash succ(Zero) : NAT}{\vdash if\ \lambda x : Bool.\ x\ then\ Zero\ else\ succ(Zero) : Nat}\ if$$

d)

$$\dfrac{\dfrac{}{\Gamma \vdash x : Bool \to Nat}\ T\text{var} \qquad \dfrac{}{\vdash y : Bool}\ T\ Var}{\underbrace{x : Bool \to Nat,\ y : Bool}_{\Gamma} \vdash x\ y : Nat}\ APP$$

Ejercicio 7 ★

Se modifica la regla de tipado de la abstracción ($\to_i$) y se la cambia por la siguiente regla:

$$\dfrac{\Gamma \vdash M : \tau}{\Gamma \vdash \lambda x\colon \sigma.\ M : \sigma \to \tau}\ \to_{i2}$$

Exhibir un juicio de tipado que sea derivable en el sistema original pero que no lo sea en el sistema actual.

$$\dfrac{\dfrac{}{\vdash x : Bool}}{\lambda x : Bool.\ x \quad : Bool \to Bool}\ \to i2$$

$$\dfrac{\dfrac{}{x : Bool \vdash x : Bool}\ T\text{Var}}{\lambda x : Bool.\ x \quad : Bool \to Bool}\ ABS$$

Ejercicio 8

Determinar qué tipo representa σ en cada uno de los siguientes juicios de tipado.

a) $\vdash$ succ(zero) $:$ σ

b) $\vdash$ isZero(succ(zero)) $:$ σ

c) $\vdash$ if (if true then false else false) then zero else succ(zero) $:$ σ

a) Nat

b) bool

c)

$$\dfrac{\dfrac{\dfrac{}{\vdash True : Bool} \quad \dfrac{}{\vdash False : Bool} \quad \dfrac{}{\vdash False : Bool}}{\vdash \text{if true then False else False} : Bool} \quad \dfrac{}{\vdash Zero : \sigma} \quad \dfrac{\dfrac{}{zero : NAT}}{\vdash Succ(zero) : \sigma}\,\text{succ}}{\vdash \text{if (if true then false else false) then zero else succ(zero)} : \sigma}$$

Tipa $\leftrightarrow$ $\sigma = NAT$

$\therefore \sigma = NAT$

Ejercicio 9 *(Tipos habitados)* ★

Decimos que un tipo τ está ***habitado*** si existe un término M tal que el juicio $\vdash M : \tau$ es derivable. En ese caso, decimos que M es un ***habitante*** de τ. Por ejemplo, dado un tipo σ, la identidad $\lambda x : \sigma . x$ es un habitante del tipo $\sigma \to \sigma$. Demostrar que los siguientes tipos están habitados (para cualquier σ, τ y ρ):

a) $\sigma \to \tau \to \sigma$

b) $(\sigma \to \tau \to \rho) \to (\sigma \to \tau) \to \sigma \to \rho$

c) $(\sigma \to \tau \to \rho) \to \tau \to \sigma \to \rho$

d) $(\tau \to \rho) \to (\sigma \to \tau) \to \sigma \to \rho$

Para pensar: el tipo b es el de la función conocida como *Combinador S*. ¿Con qué función ya conocida de Haskell se corresponden los habitantes de los otros tipos? ¿Hay tipos que no estén habitados? ¿Si se reemplaza $\to$ por $\Rightarrow$, las fórmulas habitadas son siempre tautologías? ¿Las tautologías son siempre fórmulas habitadas?

a) $\sigma \to (t \to \sigma)$ $\quad (\lambda x : \sigma . (\lambda y : t . x))$

b) $\underbrace{(\sigma \to (t \to \rho))}_{F} \to \underbrace{(\sigma \to t)}_{g} \to (\sigma \to \rho)$

x es tipo sigma, G dado σ da t y F dado σ y t da ρ

$\lambda F : \sigma \to (t \to \rho) . (\lambda G : \sigma \to t (\lambda x : \sigma . (F x (g(x)))))$

c) $(\sigma \to t \to \rho) \to \tau \to \sigma \to \rho$

$\lambda f : \sigma \to t \to \rho . \lambda x : t \; \lambda . y : \sigma . f \; y \; x$

d) $(t \to \rho) \to (\sigma \to \tau) \to \sigma \to \rho$

$\lambda f : t \to \rho . \lambda G : \sigma \to t . \lambda x : \sigma . f(G x)$

a) sería Const b) No tengo la menor idea que es Combinador S todavía

c) Es como una función flip

d) Es como el "." composición

Ejercicio 10 ★

Determinar qué tipos representan σ y τ en cada uno de los siguientes juicios de tipado. Si hay más de una solución, o si no hay ninguna, indicarlo.

a) $x:\sigma \vdash \mathsf{isZero}(\mathsf{succ}(x)) \ : \ \tau$

b) $\vdash (\lambda x:\sigma.\, x)(\lambda y: \mathsf{Bool}.\ \mathsf{zero}) \ : \ \sigma$

c) $y:\tau \vdash \mathsf{if}\ (\lambda x:\sigma.\, x)\ \mathsf{then}\ y\ \mathsf{else}\ \mathsf{succ}(\mathsf{zero}) \ : \ \sigma$

d) $x:\sigma \vdash x\ y \ : \ \tau$

e) $x:\sigma, y:\tau \vdash x\ y \ : \ \tau$

f) $x:\sigma \vdash x\ \mathsf{true} \ : \ \tau$

g) $x:\sigma \vdash x\ \mathsf{true} \ : \ \sigma$

h) $x:\sigma \vdash x\ x \ : \ \tau$

a) σ: Nat

$$\dfrac{\dfrac{\dfrac{}{x:\sigma \vdash x:\mathrm{Nat}}\ t\text{var}}{x:\sigma \vdash \mathrm{succ}(x):\mathrm{Nat}}\ \text{Tsucc}}{x:\sigma \vdash \mathrm{isZero}(\mathrm{succ}(x)):\tau}\ \text{T-isZero}$$

τ: Bool

b) ρ = NAT $\quad$ τ = Bool $\quad$ σ = Bool → NAT

$$\dfrac{\dfrac{\dfrac{}{x:\sigma \vdash x:\sigma}\ t\text{VAR}}{\vdash \lambda x:\sigma.x:\sigma\to\sigma} \quad \dfrac{\dfrac{}{y:\mathrm{Bool} \vdash \mathrm{zero}:\rho}\ \text{Tizero}}{\vdash \lambda y:\mathrm{Bool}.\mathrm{zero}:\tau\to\rho}}{\vdash (\lambda x:\sigma.x)(\lambda y:\mathrm{Bool}.\mathrm{zero}):\sigma}\ \text{Tapp}$$

$\sigma = \tau \to \rho$

c) No derivable $\quad$ $\lambda x:\sigma.x : \tau \to \rho$ pero $\neq$ Bool

$$\dfrac{y:\tau \vdash \lambda x:\sigma.x:\mathrm{Bool} \quad y:\tau \vdash y:\sigma \quad y:\tau \vdash \mathrm{succ}(\mathrm{zero}):\sigma}{y:\tau \vdash \mathrm{if}\ (\lambda x:\sigma.x)\ \mathrm{then}\ y\ \mathrm{else}\ \mathrm{succ}(\mathrm{zero}):\sigma}\ \text{Tif}$$

d)

$$\dfrac{x:\sigma \vdash x:\sigma\to\tau \quad x:\sigma \vdash y:\sigma}{x:\sigma \vdash x\,y:\tau}\ t\text{app}$$

e) $\rho:\tau$ $\sigma = (\tau \to \tau)$ con τ cualquier tipo

$$\dfrac{\dfrac{}{\Gamma \vdash x:\rho\to\tau}\ t\text{VAR} \quad \dfrac{}{\Gamma \vdash y:\rho}}{\underbrace{x:\sigma, y:\tau}_{\Gamma} \vdash x\,y:\tau}\ t\text{APP}$$

f) σ = Bool → τ

$$\dfrac{x:\sigma \vdash x:\rho\to\tau \quad x:\sigma \vdash \mathrm{true}:\rho}{x:\sigma \vdash x\ \mathrm{true}:\tau}\ \text{APP}$$

ρ = Bool

g) $\sigma = \rho \to \sigma$ ✗

$$\dfrac{\dfrac{}{x:\sigma \vdash x:\rho\to\sigma} \quad \dfrac{}{x:\sigma \vdash \mathrm{true}:\rho}\ \rho = \mathrm{Bool}}{x:\sigma \vdash x\ \mathrm{true}:\sigma}\ \text{APP}$$

h) $\sigma = \rho \to \tau$ $\quad$ $\sigma = \rho$

$$\dfrac{\dfrac{}{x:\sigma \vdash x:\rho\to\tau} \quad \dfrac{}{x:\sigma \vdash x:\rho}}{x:\sigma \vdash x\,x:\tau}$$

Ejercicio 11 *(Debilitamiento y fortalecimiento)*

Demostrar las siguientes propiedades, procediendo por inducción en la derivación del juicio correspondiente:

1. Si $\Gamma \vdash M \ : \ \sigma$ es un juicio de tipado derivable y x es una variable que no aparece en Γ, entonces $\Gamma, x:\tau \vdash M \ : \ \sigma$ es derivable para todo tipo τ. Esta regla se conoce como *debilitamiento* o *weakening*.
2. Si $\Gamma, x:\tau \vdash M \ : \ \sigma$ es un juicio de tipado derivable tal que x no aparece libre en M, entonces $\Gamma \vdash M \ : \ \sigma$ es derivable para todo tipo τ. Esta regla se conoce como *fortalecimiento* o *strengthening*.
3. Dar un contraejemplo para fortalecimiento en el caso en el que x aparece libre en M.

$\underbrace{\Gamma \vdash M:\sigma}_{\text{derivable}}$ $\quad$ $x \notin \Gamma \to \Gamma, x:\tau \vdash M:\sigma$ es derivable $\forall \tau$

Caso Base ?

Caso inductivo

Hi: $\Gamma \vdash M:\sigma$ es derivable

QVQ $\Gamma, x:\tau \vdash M:\sigma$ Pero derivar con Γ o con $\Gamma, x:\tau$ es igual si en Γ no aparece x. Es redundante

TDM

Ejercicio 12 *(Lema de sustitución ★)*

Demostrar que si valen $\Gamma, x : \sigma \vdash M : \tau$ y $\Gamma \vdash N : \sigma$ entonces vale $\Gamma \vdash M\{\mathbf{x := N}\} : \tau$.

Sugerencia: proceder por inducción en la estructura del término M.

Vale 1: $\Gamma, x : \sigma \vdash M : \tau$

Vale 2: $\Gamma \vdash N : \sigma$

HI: $\Gamma \vdash M\{x := N\} : \tau$

Como inducción en sobre M, hay que hacerlo para cada posible termino

Caso x

$$\dfrac{\overline{\Gamma \vdash N : \tau}}{\Gamma \vdash x\{x := N\} : \tau} \text{ Reemplazo}$$

Caso $(\lambda x : \tau . M)$

$$\dfrac{\overline{\Gamma \vdash \lambda x : \tau . M : \tau}}{\Gamma \vdash \lambda x : \tau . M\{x := N\} : \tau}$$

Semántica

Ejercicio 13 ★

Sean σ, τ, ρ tipos. Según la definición de sustitución, calcular:

a) $(\lambda y\colon \sigma.\, x\ (\lambda x\colon \tau.\, x))\{x := (\lambda y\colon \rho.\, x\ y)\}$

b) $(y\ (\lambda v\colon \sigma.\, x\ v))\{x := (\lambda y\colon \tau.\, v\ y)\}$

Renombrar variables en ambos términos para que las sustituciones no cambien su significado.

📌 Fundamento teórico: Sustitución segura en cálculo lambda

Cuando haces una sustitución $M\{x := N\}$, hay que tener **mucho cuidado con las variables ligadas** en M y las **variables libres** en N.

⚠️ Problema: *Captura de variables*

¿Qué es?

Una **captura de variable** ocurre si una **variable libre** del término que vas a insertar (el sustitutivo) **queda atrapada** por un λ (una ligadura) en el término original.

Ejemplo típico (con captura):

$$(\lambda x.\, y)\{y := x\} \quad \to \quad \lambda x.\, x$$

⚠️ ¡Aquí "x" era libre en el sustitutivo, pero quedó ligada por el λ!

a) $\lambda y\colon \sigma.\, x\ (\lambda z\colon \tau.\, z))\ \{x := (\lambda y\colon \rho.\, x\ y)\}$

$\lambda y\colon \sigma\ (\lambda y\colon \rho.\, x\ y)(\lambda z\colon \tau.\, z))$

Renombro variables ligadas

✅ Solución: *Renombrar variables ligadas (α-conversión)*

Para evitar esto, **antes de sustituir**, debes revisar:

> Si alguna variable **ligada** en el término original **coincide con alguna variable libre** en el término sustitutivo, **renómbrala.**

🧠 Regla práctica:

- **Paso 1**: Identifica todas las **variables ligadas** en el término original M.
- **Paso 2**: Identifica todas las **variables libres** en el sustitutivo N.
- **Paso 3**: Si alguna variable ligada en M también aparece libre en N, **renómbrala** en M antes de hacer la sustitución.

b) $(y\ (\lambda z\colon \sigma.\, x\ z))\ \{x := (\lambda y\colon \tau.\, v\ y)\}$

$y\ (\lambda z\colon \sigma.\, (\lambda y\colon \tau.\, v\ y)\ z)$

Ejercicio 14 *(Conmutación de sustituciones)*

Sean M, N y P términos del cálculo-λ.

a) Por inducción en la estructura del término M, demostrar que si x no aparece libre en P y $x \neq y$, entonces:

$$M\{x := \mathbf{N}\}\{y := \mathbf{P}\} = M\{y := \mathbf{P}\}\{x := \mathbf{N}\{y := \mathbf{P}\}\}$$

b) Dar un contraejemplo para la ecuación de arriba cuando x aparece libre en P.

Caso x

Ejercicio 15 *(Valores)* ★

Dado el conjunto de valores visto en clase:

$$V := \lambda x : \tau.\ M \mid \mathsf{true} \mid \mathsf{false} \mid \mathsf{zero} \mid \mathsf{succ}(V)$$

Determinar si cada una de las siguientes expresiones es o no un valor:

a) $(\lambda x : \mathsf{Bool}.\ x)\ \mathsf{true}$ ✗

b) $\lambda x : \mathsf{Bool}.\ \underline{2}$ ✓

c) $\lambda x : \mathsf{Bool}.\ \mathsf{pred}(\underline{2})$ ✓

d) $\lambda y : \mathsf{Nat}.\ (\lambda x : \mathsf{Bool}.\ \mathsf{pred}(\underline{2}))\ \mathsf{true}$ ✗

e) x ✗

f) $\mathsf{succ}(\mathsf{succ}(\mathsf{zero}))$ ✓

Ejercicio 16 *(Programa, Forma Normal)* ★

Para el siguiente ejercicio, considerar el cálculo **sin** la regla $\mathsf{pred}(\mathsf{zero}) \to \mathsf{zero}$

Un ***programa*** es un término que tipa en el contexto vacío (es decir, no puede contener variables libres). Para cada una de las siguientes expresiones,

a) Determinar si puede ser considerada un **programa**.

b) Si es un programa, ¿Cuál es el resultado de su evaluación? Determinar si se trata de una forma normal, y en caso de serlo, si es un **valor** o un **error**.

I $(\lambda x : \mathsf{Bool}.\ x)\ \mathsf{true}$

II $\lambda x : \mathsf{Nat}.\ \mathsf{pred}(\mathsf{succ}(x))$

III $\lambda x : \mathsf{Nat}.\ \mathsf{pred}(\mathsf{succ}(y))$

IV $(\lambda x : \mathsf{Bool}.\ \mathsf{pred}(\mathsf{isZero}(x)))\ \mathsf{true}$

V $(\lambda f : \mathsf{Nat} \to \mathsf{Bool}.\ f\ \mathsf{zero})\ (\lambda x : \mathsf{Nat}.\ \mathsf{isZero}(x))$

VI $(\lambda f : \mathsf{Nat} \to \mathsf{Bool}.\ x)\ (\lambda x : \mathsf{Nat}.\ \mathsf{isZero}(x))$

VII $(\lambda f : \mathsf{Nat} \to \mathsf{Bool}.\ f\ \mathsf{pred}(\mathsf{zero}))\ (\lambda x : \mathsf{Nat}.\ \mathsf{isZero}(x))$

VIII $\mu y : \mathsf{Nat}.\ \mathsf{succ}(y)$

1. $(\lambda x : Bool.\ x)\ True$ es programa

$(\lambda x : Bool.\ x)\ True$

$\xrightarrow{\beta}$ $True$ es valor

2. $\lambda x : Nat.\ pred(succ(x))$ es programa, es valor

3. $\lambda x : Nat.\ pred(succ(y))$ No es programa

4. $(\lambda x : Bool.\ pred(isZero(x)))\ True$ es programa

$\xrightarrow{\beta}$ $Pred(isZero(True))$ Error! Es Forma Normal

V. $(\lambda f : Nat \to Bool.\ f\ Zero)(\lambda x : Nat.\ isZero(x))$ P

$\xrightarrow{\beta}$ $(\lambda x : Nat.\ isZero(x))\ Zero$

$\xrightarrow{\beta}$ $isZero(Zero) \xrightarrow{zero} true$ valor

VI. $(\lambda f : Nat \to Bool.\ x)\ (\lambda x : Nat.\ isZero(x))$ No P

VII. $(\lambda f : Nat \to Bool.\ f\ pred(Zero))\ (\lambda x : Nat\ isZero(x))$ P

$\xrightarrow{\beta}$ $(\lambda x : Nat.\ isZero(x))\ pred(Zero))$

$\xrightarrow{\beta}$ $isZero(pred(Zero))$ Forma Normal, error.

VIII. $\mu y : Nat.\ Succ(y)$

Aun no lo puedo hacer

Ejercicio 17 *(Determinismo)*

a) ¿Es cierto que la relación definida $\to$ está determinada (es una función parcial)? Más precisamente, ¿pasa que si $M \to N$ y $M \to N'$ entonces también vale $N = N'$? → Sí !!

b) ¿Vale lo mismo con muchos pasos? Es decir, ¿es cierto que si $M \twoheadrightarrow M'$ y $M \twoheadrightarrow M''$ entonces $M' = M''$? → AJA

c) ¿Acaso es cierto que si $M \to M'$ y $M \twoheadrightarrow M''$ entonces $M' = M''$?

→ No necesariamente .. Mas Bien diría No

Ejercicio 18

a) ¿Da lo mismo evaluar $\mathsf{succ}(\mathsf{pred}(M))$ que $\mathsf{pred}(\mathsf{succ}(M))$? ¿Por qué? → No porque las reglas que tenemos en Pred Succ sino usan las que reducen el interior Pero!

$succ(\underbrace{pred(Zero)}_{\times})$ $pred(succ(Zero))$ ✓

b) ¿Es verdad que para todo término M vale $\mathsf{isZero}(\mathsf{succ}(M)) \twoheadrightarrow \mathsf{false}$? Si no lo es, ¿para qué términos vale?

Vale siempre, yo asumo que vale incluso con un $pred(Zero)$ xq la regla en M (para Todo Término) y $pred(Zero)$ es Un Término.

Pero la regla real es ... $\underline{N}$ y ahí no vale siempre xq $pred(Zero)$ no se puede escribir como $\underline{N}$, es Una forma normal entonces y daría error

c) ¿Para qué términos M vale $\mathsf{isZero}(\mathsf{pred}(M)) \twoheadrightarrow \mathsf{true}$? (Hay infinitos).

Todo termino $\neq Zero$

o con la otra def

Todo termino

de la forma $succ^{n}(Zero)$ $n \geq 1$

Ejercicio 19

Al agregar la siguiente regla para las abstracciones:

$$\textbf{Si } M \to M', \textbf{ entonces: } \lambda x\colon \tau.\ M \to \lambda x\colon \tau.\ M' \qquad (\xi)$$

a) Repensar el conjunto de valores para respetar esta modificación, pensar por ejemplo si $(\lambda x\colon \mathsf{Bool}.\ (\lambda y\colon \mathsf{Bool}.\ y)\ \mathsf{true})$ es o no un valor.

b) ¿Qué reglas deberían modificarse para no perder el determinismo?

c) Utilizando el cálculo modificado y los valores definidos, reducir la siguiente expresión $(\lambda x\colon \mathsf{Nat} \to \mathsf{Nat}.\ x\ \underline{23})\ (\lambda x\colon \mathsf{Nat}.\ \mathsf{pred}(\mathsf{succ}(\mathsf{zero})))$
¿Qué se puede concluir entonces? ¿Es una buena idea agregar esta regla?

a) No sería, xq si los valores tienen que estar en la máxima reducción posible, puedes aplicar β adentro

$\xrightarrow{\xi\beta} (\lambda x: Bool.\ True)$

El valor sería $(\lambda x:\sigma.\ F)$ siendo F una Forma Normal

b) Ahora hay 2 caminos. Sustituir y reducir

Para evitar eso buscamos que primero reduzcamos y luego sustituir: $(\lambda x:\sigma.\ F)\ V \to F\{x:=V\}$

c) $(\lambda x:Nat\to Nat.\ x\ \underline{23})(\lambda x:Nat.\ pred(succ(Zero)))$

$\xrightarrow{\xi} (\lambda x:Nat\to Nat.\ x\ \underline{23})(\lambda x:Nat.\ Zero)$

$\xrightarrow{\beta} (\lambda x:Nat.\ Zero)\ \underline{23} \xrightarrow{\beta} Zero$

Honestamente, no sé, son 2 enfoques de eval diferente. Este puede encontrar errores antes, pero también calcular cosas que no van a ser usadas.

Extensiones

En esta sección puede asumirse, siempre que sea necesario, que el cálculo ha sido extendido con la suma de números naturales $(M + N)$, con las siguiente reglas de tipado y semántica:

$$\frac{\Gamma \vdash M : \mathsf{Nat} \quad \Gamma \vdash N : \mathsf{Nat}}{\Gamma \vdash M + N : \mathsf{Nat}} +$$

Si $M \to M'$, **entonces:** $M + N \to M' + N$	$(+_{c1})$
Si $N \to N'$, **entonces:** $V + N \to V + N'$	$(+_{c2})$
$V + \mathsf{zero} \to V$	$(+_0)$
$V_1 + \mathsf{succ}(V_2) \to \mathsf{succ}(V_1) + V_2$	$(+_{\mathsf{succ}})$

Ejercicio 20 *(Pares, o productos)* ★

Este ejercicio extiende el cálculo-λ tipado con *pares*. Las gramáticas de los tipos y los términos se extienden de la siguiente manera:

$$\begin{array}{rcl} \tau & ::= & \ldots \mid \tau \times \tau \\ M & ::= & \ldots \mid \langle M, M\rangle \mid \pi_1(M) \mid \pi_2(M) \end{array}$$

donde $\sigma \times \tau$ es el tipo de los pares cuya primera componente es de tipo σ y cuya segunda componente es de tipo τ, $\langle M, N\rangle$ construye un par y $\pi_1(M)$ y $\pi_2(M)$ proyectan la primera y la segunda componente de un par, respectivamente.

a) Definir reglas de tipado para los nuevos constructores de términos.

b) Usando las reglas de tipado anteriores, y dados los tipos σ, τ y ρ, exhibir habitantes de los siguientes tipos:

- I) Constructor de pares: $\sigma \to \tau \to (\sigma \times \tau)$
- II) Proyecciones: $(\sigma \times \tau) \to \sigma$ y $(\sigma \times \tau) \to \tau$
- III) Conmutatividad: $(\sigma \times \tau) \to (\tau \times \sigma)$,
- IV) Asociatividad: $((\sigma \times \tau) \times \rho) \to (\sigma \times (\tau \times \rho))$ y $(\sigma \times (\tau \times \rho)) \to ((\sigma \times \tau) \times \rho)$.
- V) Currificación: $((\sigma \times \tau) \to \rho) \to (\sigma \to \tau \to \rho)$ y $(\sigma \to \tau \to \rho) \to ((\sigma \times \tau) \to \rho)$.

a)

$$\frac{\Gamma \vdash M : t \qquad \Gamma \vdash N : \sigma}{\Gamma \vdash \langle M, N\rangle : \tau\times\sigma}\ PAR \qquad \frac{\Gamma \vdash M : \sigma\times t}{\Gamma \vdash \pi_1(M) : \sigma} \qquad \frac{\Gamma \vdash M : t\times\sigma}{\Gamma \vdash \pi_2(M) : \sigma}$$

b)

I) $\sigma \to t \to (\sigma \times \tau)$

$\lambda x:\sigma.\ \lambda y:\sigma.\ \langle x, y\rangle$

II) $(\sigma\times\tau) \to \sigma$ y $(\sigma\times\tau)\to t$

$\lambda.P:(\sigma\times t).\ \pi_1(P)$ y $\lambda.P:(\sigma\times t).\ \pi_2(P)$

III) $(\sigma\times\tau) \to (t\times\sigma)$

$\lambda P:(\sigma\times t).\ \langle \pi_2(P), \pi_2(P)\rangle$

IV) $((\sigma\times t)\times P) \to (\sigma\times(t\times P))$

$\lambda Q((\sigma\times t)\times P).($

$\langle \pi_1(\pi_1(Q), \langle \pi_2(\pi_1(Q)), \pi_2(Q)\rangle\rangle)$

V) $\lambda f:(\sigma\times\tau\to P).\lambda x:\sigma.\lambda y:t.\ f\ x\ y$

$\lambda f:(\sigma\to t\to P).\ \lambda Q:(\sigma\times t).\ f\ \pi_1(Q)\ \pi_2(Q)$

c) ¿Cómo se extiende el conjunto de los valores? ---------> $V ::= \ldots \mid \langle V, V \rangle$

d) Definir reglas de semántica operacional manteniendo el determinismo y la preservación de tipos. **Importante:** no olvidar las reglas de congruencia.

e) Demostrar el determinismo de la relación de reducción definida. ¿Se verifica la propiedad de preservación de tipos? ¿Se verifica la propiedad de progreso?

d) $\pi_1(\langle V, V' \rangle) \xrightarrow{\pi_1} V$ $\quad$ $\pi_2(\langle V, V' \rangle) \xrightarrow{\pi_2} V'$

si $M \to M'$: $\langle M, N \rangle \to \langle M', N \rangle$ $\quad$ A)

si $M \to M'$: $\langle V, M \rangle \to \langle V, M' \rangle$ $\quad$ B)

si $M \to M'$: $\pi_{1/2}(M) \to \pi_{1/2}(M')$ $\quad$ c)

e) determinismo

Sumo a lo dicho de clase xq todo lo puedo suponer que no tiene "los puntos"

Caso Base

Caso π_1 : $M \to M_1$ entonces $M = \pi_1(\langle V, V' \rangle)$ $\quad$ $M_1 = V$

veo que no hay otra regla posible xq no cumplen los puntos, y es π_1 no π_2. ∴ $M \to M_2$

Usando si π_1 ∴ $M_1 = M_2$

Caso π_2 : Análogo

Caso inductivo

HI: $M' \to M_1'$, $M' \to M_2' \Rightarrow M_1' = M_2'$

Caso A $\quad$ $M \to M_1$ $\quad$ $M = \langle M', N \rangle$ $\quad$ $M_1 = \langle M_1', N \rangle$ donde $M' \to M_1'$ No aplica otra...

Caso B $\quad$ $M \to M_1$ $\quad$ $M = \langle V, M' \rangle$ $\quad$ $M_1 = \langle V, M_1' \rangle$ donde $M' \to M_1'$ No aplica otra

Caso C_I $\quad$ " $\quad$ $M = \pi_1(M')$ $\quad$ $M_1 = \pi_1(M_1')$ $\quad$ " $\quad$ "

Caso C_{II} $\quad$ " $\quad$ $M = \pi_{11}(M')$ $\quad$ $M_1 = \pi_{11}(M_1')$ $\quad$ " $\quad$ "

si no aplica otra $M \to M_2$ $\quad$ $M_1 = M_2$ para cada caso.