

Práctica 1

1. Lambda cálculo sin tipos

1. ¿Cuáles de las siguientes expresiones son correctas en cálculo lambda según la gramática? En caso de que alguna expresión no lo sea, explicar porqué, y en caso de que sí lo sea, explicar a qué corresponde (una función, una aplicación, etc).

a) λxx

g) $\lambda x.y\lambda z.z$

b) $xyz(yx)$

h) $vx(yz)(\lambda w.wy)$

c) $\lambda x.yxz$

i) $(\lambda xyz.xz(yz))uvw$

d) $\lambda x.x(\lambda y.z)$

j) $\text{match}(\lambda x.x, x.x, y.x)$

e) $x\lambda y$

k) $\text{match}((\lambda x.x)\text{inl}(\star), x.x, y.y)$

f) $(\lambda x.yxx)zv$

l) $(\lambda x.x); \star$

2. Reducir las siguientes expresiones

a) $(\lambda x.\lambda y.xy)(\lambda x.xy)$

b) $(\lambda x.\lambda y.xy)(\lambda x.xy)y$

c) $(\lambda z.\lambda y.y(\lambda x.x)z)\lambda x.xy$

d) $(\lambda x.\text{match}(\text{inl}(x), x.\lambda y.x, y.\lambda y.y))\lambda x.x$

e) $(\lambda x.\text{match}(x, x.\lambda y.x, y.\lambda y.y))\pi_2\langle \star, \lambda x.x \rangle$

f) $(\lambda x.\text{err}(x))\langle \top, \perp \rangle$

3. Escribir un programa que tome dos booleanos (con la codificación dada al final de la sección 4.2.3) y devuelva el resultado del XOR entre ellos.

2. Estrategias de reducción

4. ¿Cuales de las siguientes reducciones son correctas?

a) $\lambda x.x \longrightarrow \lambda x.x$

b) $\lambda x.x \longrightarrow^* \lambda x.x$

c) $\lambda x.x \longrightarrow^+ \lambda x.x$

d) $(\lambda x.\lambda y.x)tu \longrightarrow t$

e) $(\lambda x.\lambda y.x)tu \longrightarrow^* t$

f) $(\lambda x.x)((\lambda y.t)u) \longrightarrow (\lambda x.x)t$ con $y \notin FV(t)$

g) $(\lambda x.x)((\lambda y.t)u) \longrightarrow^* (\lambda x.x)t$ con $y \notin FV(t)$

h) $(\lambda x.x)((\lambda y.t)u) \longrightarrow^+ (\lambda x.x)t$ con $y \notin FV(t)$

i) $(\lambda x.xx\star)(\lambda x.xx\star) \longrightarrow (\lambda x.xx\star)(\lambda x.xx\star) \star \star$

$$j) (\lambda x.xx\star)(\lambda x.xx\star) \longrightarrow^* (\lambda x.xx\star)(\lambda x.xx\star) \star \star$$

$$k) (\lambda x.xx\star)(\lambda x.xx\star) \longrightarrow^+ (\lambda x.xx\star)(\lambda x.xx\star) \star \star$$

5. Determinar todos los redexes de cada término.

$$a) (\lambda x.x)((\lambda x.x)\lambda x.x)$$

$$b) (\lambda x.\lambda y.(\lambda z.z)x)\Omega_\star$$

$$c) \text{match}(\text{inl}(\star), x.(\lambda x.x)\star, y.\text{inl}(\star))$$

$$d) \lambda x.(\lambda y.(\lambda z.(\lambda z.z)z)y)x$$

6. Dar todas las reducciones posibles de los siguientes términos.

$$a) (\lambda x.\lambda y.yy)(\Omega_\star)\lambda x.x$$

$$b) \lambda x.(\lambda y.(\lambda z.(\lambda w.w)z)y)x$$

7. ¿Cuál de los siguientes tiene forma normal?

$$a) (\lambda x.x)((\lambda x.\lambda y.x)(\lambda y.y)(\lambda z.zz))$$

$$b) (\lambda x.xx)(\lambda y.y)(\lambda x.xx)$$

$$c) (\lambda x.\lambda y.xyy)(\lambda y.y)(\lambda x.xx)$$

8. Dar dos términos que cumplan:

a) Están en forma normal.

b) No están en forma normal pero son fuertemente normalizables.

c) Normalizables pero no fuertemente normalizables.

d) No normalizables

9. Demostrar que un término es cerrado (es decir, sin variables libres) y en forma normal en una reducción débil sí y sólo sí tiene la forma:

- \star
- $v; u$, donde v y u son cerrados y en forma normal y $v \neq \star$.
- $\text{err}(v)$, donde v es cerrado y en forma normal.
- $\lambda x.t$, con $FV(t) \subseteq \{x\}$.
- vu , donde v y u son cerrados y en forma normal y v no es de la forma $\lambda x.t$.
- $\langle v, u \rangle$, donde v y u son cerrados y en forma normal.
- $\pi_1 v$, donde v es cerrado y en forma normal y no es de la forma $\langle v_1, v_2 \rangle$.
- $\pi_2 v$, donde v es cerrado y en forma normal y no es de la forma $\langle v_1, v_2 \rangle$.
- $\text{inl}(v)$, donde v es cerrado y en forma normal.
- $\text{inr}(v)$, donde v es cerrado y en forma normal.
- $\text{match}(v, x.t, y.r)$ donde v es cerrado y en forma normal y no es de la forma $\text{inl}(v')$ ni $\text{inr}(v')$.

Sugerencia: Proceder por inducción estructural en los términos.

10. ¿Cuales son los términos cerrados y en forma normal en call-by-name? ¿y en call-by-value?
11. Dar la traza de $(\lambda x.\langle x, x \rangle)((\lambda x.x)\star; \text{inl}(\star))$ en call-by-name y en call-by-value.
12. Dar la traza de $(\lambda x.(\lambda x.\star)(\star; \langle x, x \rangle))((\lambda x.yxx)\lambda x.xx)$ en
 - a) call-by-name fuerte
 - b) call-by-name débil
 - c) call-by-value fuerte
 - d) call-by-value débil

3. Lambda cálculo tipado

13. Tipar los siguientes términos (si es posible).
 - a) $\lambda x.x$
 - b) $\lambda x.\lambda y.xy$
 - c) $\lambda x.\lambda y.xy$
 - d) Los términos del ejercicio 5.
14. Extender el cálculo lambda extendido con booleanos primitivos `true`, `false` y una construcción `if – then – else`.
 - a) Dar la gramática, semántica operacional y reglas de tipado.
 - b) Tipar:
 - 1) `if $(\lambda x.x)\text{true}$ then $\text{inl}(\star)$ else $\text{inr}(\star)$`
 - 2) `$(\lambda x.\text{if } x \text{ then } (\lambda x.x) \text{ else } \text{err}(\star))\text{false}$`
 - 3) `$\lambda x.\text{if } x \text{ then false else true}$`
15. Extender el cálculo lambda extendido con constructores de listas de elementos de tipo $\top \vee \top$:

<code>nil</code>	(lista vacía)
<code>cons tu</code>	(agregar el elemento t a la lista u)
<code>if $\text{nil } t \text{ then } u \text{ else } v$</code>	(if lista vacía)
<code>hd t</code>	(head)
<code>tl t</code>	(tail)

- a) Dar la gramática, semántica operacional y reglas de tipado.
- b) Escribir una función haga el `And` de todos los elementos de una lista de 4 elementos. Tiparla.
- c) Tipar la función del ítem anterior aplicada a

`cons $\text{inl}(\star)(\text{cons } ((\lambda x.\text{inr}(\star))\text{inl}(\star))(\text{cons } \text{inr}(\star)(\text{cons } \text{inl}(\star) \text{ nil})))$`

y dar su traza de reducción en CBN.