Práctica 3: Lambda cálculo y teoría de tipos

1. PCF no tipado

1. ¿Cuáles de las siguientes expresiones son correctas en PCF 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
   b) $xyz(yx)$
   c) $\lambda x.yxz$
   d) $\lambda x.x(\lambda y.z)$
   e) $x\lambda y$
   f) $(\lambda x.yxx)zv$
   g) $\lambda x.y\lambda z.z$
   h) $vx(yz)(\lambda w.wy)$
   i) $(\lambda xyz.xz(yz))uvw$
   j) $\mu x\lambda y.y$
   k) $\mu x.\lambda x.x$
   l) $\mu x.x$
   m) $\lambda x.x?1:2$
   n) $(\lambda x.x-1)1?1:2$
   ñ) $\mu?1:(\lambda x.x)$
   o) $(\lambda x.x)/0$
   p) $2/1$
   q) $\lambda x.t?r:su$

2. Reducir las siguientes expresiones

   a) $(\lambda x.x)\lambda x.x$
   b) $(\lambda x.\lambda y.xy)\lambda x.xy$
   c) $(\lambda x.\lambda y.xy)(\lambda x.xy)y$
   d) $(\lambda z.\lambda y.y(\lambda x.x)z)\lambda x.xy$
   e) $\mu x.\lambda y.y$
   f) $(\lambda x.x?\lambda y.x:\lambda y.y)0\ 1$
   g) $(\mathsf{let}\ x=0\ \mathsf{in}\ x?\lambda y.x:\lambda y.y)1$
   h) $\mathsf{let}\ y=(\lambda x.\mathsf{let}\ y=x\ \mathsf{in}\ yy)\ \mathsf{in}\ yy$

3. Dar la traza de $(\mu f.\lambda n.n?1:n\times(f(n-1)))3$.

4. Escribir un programa en PCF que tome dos enteros n y p y calcule n^p. Dar la traza de 3^1 y 3^2.

5. Escribir un programa en PCF que tome un entero n y devulva 1 si es un número primo o 0 sino. Dar la traza de PRIMO 4 y PRIMO 3.

2. Estrategias de reducción

6. ¿Cuál de las siguientes reducciones es verdadera?

   a) $\lambda x.x\to\lambda x.x$
   b) $\lambda x.x\to^*\lambda x.x$
   c) $\lambda x.x\to^+\lambda x.x$
   d) $(\lambda x.\lambda y.x)tu\to t$
   e) $(\lambda x.\lambda y.x)tu\to^* t$
   f) $\mu x.x\to\mu x.x$
   g) $\mu x.x\to^*\mu x.x$
   h) $\mu x.x\to^+\mu x.x$
   i) $(\lambda x.x)((\lambda y.t)u)\to(\lambda x.x)t$ con $y\notin FV(t)$
   j) $(\lambda x.x)((\lambda y.t)u)\to^*(\lambda x.x)t$ con $y\notin FV(t)$
   k) $(\lambda x.x)((\lambda y.t)u)\to^+(\lambda x.x)t$ con $y\notin FV(t)$

7. Determinar todos los redexes de cada término.

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

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

c) $2 - 1?(\lambda x.x)2 : 1$

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

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

   a) $(\lambda x.\lambda y.yy)(\mu x.x)\lambda x.x$

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

9. ¿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) $(\lambda x.\lambda y.xyy)(\lambda y.y)(\lambda x.xx)$

10. 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

11. 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:

    - $\lambda x.t$, con $FV(t) = \{x\}$ o $FV(t) = \emptyset$
    - $n \in \mathbb{N}$
    - uv donde u y v son cerrados e irreducibles y u no es de la forma $\lambda x.t$.
    - $u \otimes v$ donde u y v son cerrados e irreducibles y no ambos son constantes numéricas, o $u \in \mathbb{N}$, $\otimes = /$ y $v = 0$.
    - $u?v : s$, donde u es cerrado e irreducible y no es una constante numérica y $FV(v) = FV(s) = \emptyset$.

12. ¿Cuales son los términos cerrados y en forma normal en call-by-name? ¿y en call-by-value?

13. Dar la traza de $(\lambda x.x \times x)(5 + 6)$ en call-by-name y en call-by-value.

14. Dar la traza de $(\lambda x.(\lambda x,0)(x + x))(\mathit{Fact}\ 2)$ en

    a) call-by-name fuerte

    b) call-by-name débil

    c) call-by-value fuerte

    d) call-by-value débil

15. Dar la traza de $(\lambda x.\lambda y.(1 + x) + y)((\lambda z.z \times z)2)$ en

a) call-by-name fuerte
b) call-by-name débil
c) call-by-value fuerte
d) call-by-value débil

3. PCF tipado

16. Tipar los siguientes términos (si es posible).

    a) $\lambda x : \mathsf{nat}.x$
    b) $\lambda x : \mathsf{nat} \Rightarrow \mathsf{nat}.x$
    c) $\lambda x : \mathsf{nat} \Rightarrow \mathsf{nat}.\lambda y : \mathsf{nat} \Rightarrow \mathsf{nat}.xy$
    d) $\lambda x : \mathsf{nat} \Rightarrow \mathsf{nat}.\lambda y : \mathsf{nat}.xy$
    e) $\mathsf{let}\ x : \mathsf{nat} \Rightarrow \mathsf{nat} = \lambda x : \mathsf{nat}.x + 1\ \mathsf{in}\ x3$
    f) $\mathsf{let}\ x : \mathsf{nat} \Rightarrow \mathsf{nat} = \lambda x : \mathsf{nat}.x + 1\ \mathsf{in}\ x(\lambda x.x)$
    g) $\mu f : A.\lambda n : B.\lambda m : C.m?1 : n \times fn(m-1)$
    para algún A, B y C. ¿De qué función se trata?

17. Extender PCF con booleanos true, false e $\mathsf{if} - \mathsf{then} - \mathsf{else}$.

    a) Dar la gramática, semántica operacional y reglas de tipado.
    b) Tipar:
        1) $\mathsf{if}\ (\lambda \mathsf{x} : \mathsf{A.x})\mathsf{true}\ \mathsf{then}\ 2\ \mathsf{else}\ 1$, para algún A.
        2) $(\lambda x : \mathsf{bool.if}\ \mathsf{x}\ \mathsf{then}\ (\lambda \mathsf{x} : \mathsf{nat.x})\ \mathsf{else}\ 0)\mathsf{false}$
        3) $\lambda x : \mathsf{bool.if}\ \mathsf{x}\ \mathsf{then}\ \mathsf{false}\ \mathsf{else}\ \mathsf{true}$

18. Extender PCF con constructores para pares: (t, u), $\pi_1(t, u)$ y $\pi_2(t, u)$.

    a) Dar la gramática, semántica operacional y reglas de tipado.
    b) Escribir una suma que reciba un par y otra que reciba dos argumentos, y tiparlos.
    c) Definir una función curry que tome una función que espera un par, y dos argumentos, y ejecute esa función con esos argumentos y tiparla.
    d) Definir una función uncurry que tome una función que espera dos argumentos y un par, y ejecute esa función con los elementos del par, y tiparla.

19. Extender PCF con constructores de listas de números:

nil	(lista vacía)
$\mathsf{cons}\ tu$	(agregar el elemento t a la lista u)
$\mathsf{ifnil}\ t\ \mathsf{then}\ u\ \mathsf{else}\ v$	(if lista vacía)
$\mathsf{hd}\ t$	(head)
$\mathsf{tl}\ t$	(tail)

    a) Dar la gramática, semántica operacional y reglas de tipado.

b) Escribir una función sume los elementos de una lista de 4 elementos. Tiparla.

c) Tipar la función del ítem anterior aplicada a

$$\mathsf{cons}\ 0(\mathsf{cons}\ ((\lambda \mathsf{x} : \mathsf{nat}.\mathsf{x} + 1)2)(\mathsf{cons}\ 2(\mathsf{cons}\ 1\ \mathsf{nil})))$$

y dar su traza de reducción en CBN.

4. Inferencia de tipos

20. Dar el tipo principal, y todos los tipos, de los siguientes términos.

a) $\lambda x.x$

b) $(\lambda x.x + 1)2$

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

d) $\mathsf{let}\ x = \lambda x.x + 1\ \mathsf{in}\ x3$

e) $\mathsf{let}\ x = \lambda x.x + 1\ \mathsf{in}\ x(\lambda x.x)$

f) $\mu f.\lambda n.\lambda m.m?1 : n \times fn(m-1)$

g) $\lambda x.\lambda y.x(y + 1) + 2$

h) $\mathsf{let}\ x = \lambda x.x\ \mathsf{in}\ xx$

i) $(\lambda x.x)(\lambda x.x)$

j) $\lambda x.xx$

21. Extender los algoritmos de Hindley y Robinson para la extensión de PCF con booleanos del ejercicio 17.

22. Extender los algoritmos de Hindley y Robinson para la extensión de PCF con pares del ejercicio 18.

23. Extender los algoritmos de Hindley y Robinson para la extensión de PCF con listas del ejercicio 19.

24. Extender PCF con árboles binarios, extendiendo: la gramática, la semántica operacional, las reglas de tipado y el algoritmo de inferencia de tipos. Los constructores deberían ser:

$\mathsf{L}n$: Una hoja con contenido n

$\mathsf{T}ntu$: Un árbol binario, con subárboles t y u y n en su nodo

$\mathsf{ifLeaf}\ t\ \mathsf{then}\ u\ \mathsf{else}\ v$ Testea si es una hoja

$\mathsf{content}\ t$: Contenido del nodo

$\mathsf{left}\ t$: Subárbol izquierdo (reduce a t si es una hoja)

$\mathsf{right}\ t$: Subárbol derecho (reduce a t si es una hoja)

25. En el caso general de un lenguaje sin variables ligadas, reemplazamos las primeras tres reglas del algoritmo de Robinson por las siguientes dos:

- Si una ecuación tiene la forma

$$f(U_1, \ldots, U_n) = f(V_1, \ldots, V_n)$$

reemplazarla por las ecuaciones $U_1 = V_1, \ldots, U_n = V_n$.

- Si una ecuación tiene la forma

$$f(U_1, \ldots, U_n) = g(V_1, \ldots, V_n)$$

  done f y g son símbolos distintos, devolver error.

a) En un lenguaje formado por un símbolo $+$ que toma dos argumentos y las constantes enteras:

1) ¿Tiene solución la ecuación $(2 + (3 + X)) = (X + (Y + 2))$?
2) ¿Tiene solución la ecuación $X + 2 = 4$?
3) ¿Cuál es la diferencia entre las dos ecuaciones anteriores y lo que aprendimos en la escuela?

b) Usar la semántica operacional de PCF para definir otra noción de solución que coincida con lo que vimos en la escuela, y mostrar que $X + 2 = 4$ tiene solución en ese caso.

5. Polimorfismo

26. Dar el tipo más general a cada uno de los términos del ejercicio 20.

27. Extender PCF polimórfico con listas, como en el ejercicio 19, sólo que ahora usaremos el tipo list con un tipo como argumento. Así, escribimos $\mathsf{list}[\mathsf{nat}]$ al tipo de listas de naturales, $\mathsf{list}[\mathsf{nat} \Rightarrow \mathsf{nat}]$ al tipo de las listas de funciones de naturales en naturales, y $\mathsf{list}[\mathsf{list}[\mathsf{nat}]]$ al tipo de las listas que como elementos tienen listas de naturales.

    Escribir la función map que toma una función f y una lista, y aplica f a cada elemento de la lista. Dar un tipo para map

28. System F es un sistema de tipos polimórfico más general que el visto en clase. En System F el polimorfismo puede aparecer en cualquier lugar, sin restricciones (a excepción de la restricción de la introducción del cuantificador universal, que implide derivar el tipo $\forall X.X$). En consecuencia, la inferencia se vuelve indecidible.

    La gramática de tipos de System F es la siguiente:

$$A ::= X \mid \mathsf{nat} \mid A \Rightarrow A \mid \forall X.A$$

a) Dar las reglas de tipado de System F.

b) Dar los tipos más generales, en System F, para los términos del punto 20.

6. Interpretación

29. Evaluar en el intérprete CBN las siguientes expresiones.

a) $(\lambda x.x)0$

b) $(\lambda x.\lambda x.x)23$

c) $(\lambda x.\lambda y.(\lambda x.x + y)x)54$

d) $\mathsf{let}\ x = 4\ \mathsf{in}\ (\mathsf{let}\ f = (\lambda y.y + x)\ \mathsf{in}\ (\mathsf{let}\ y = 5\ \mathsf{in}\ f6))$

30. Extender el intérprete CBN para pares.

31. Extender el intérprete CBN para booleanos.

32. Evaluar en los intérprete CBN y CBV las siguientes expresiones.

    a) $(\lambda x.x + x)((\lambda y.y)3)$

    b) $(\lambda x.x + x)((\mu f.\lambda n.n?1 : n \times f(n-1))2)$

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

33. Extender el intérprete CBV para pares.

34. Extender el intérprete CBV para booleanos.

7. Semántica denotacional

35. Dar la semántica denotacional de los siguientes términos.

    a) $\lambda x : \mathsf{nat}, 0$

    b) $\mu x : \mathsf{nat}.x$

    c) $\lambda x : \mathsf{nat}.(\lambda y : \mathsf{nat}.y)x + 1$

    d) $(\lambda x : \mathsf{nat}, 0)(\mu x : \mathsf{nat}.x)$

    e) Fact

    f) $\mathsf{Fact}\ 2$

36. Si $[\![t]\!]_\theta = 0$, $[\![r]\!]_\theta = 0$ y $[\![s]\!]_\theta = \bot_{\mathbb{N}}$, ¿quién es $[\![t?r : s]\!]_\theta$?

37. Dar la semántica operacional a grandes pasos, a pequeños pasos y la denotacional a los siguientes términos:

    a) $\lambda x : \mathsf{nat}.x$

    b) $\lambda x : \mathsf{nat} \Rightarrow \mathsf{nat}.x$

    c) $\lambda x : \mathsf{nat} \Rightarrow \mathsf{nat}.\lambda y : \mathsf{nat}.xy$

    d) $\mathsf{let}\ x : \mathsf{nat} \Rightarrow \mathsf{nat} = \lambda x : \mathsf{nat}.x + 1\ \mathsf{in}\ x3$

    e) $(\lambda x : \mathsf{nat}.\mu x : \mathsf{nat}.x + 1)((\lambda x : \mathsf{nat}.x)2)$

    f) $\mu f : \mathsf{nat} \Rightarrow \mathsf{nat} \Rightarrow \mathsf{nat}.\lambda n : \mathsf{nat}.\lambda m : \mathsf{nat}.m?1 : n \times fn(m-1)$

38. Modificar la semántica denotacional incluyendo el elemento error en todo conjunto, con el fin de detectar la división por 0.

39. Extender la semántica denotacional con error, para PCF con pares.

40. Probar el teorema 4.45.

41. Probar el teorema de soundness (4.46).

8. Intro a lógica lineal

42. Mostrar que cada una de las reglas resultantes de ignorar los términos en la relación de tipado de la Sección 4.8.3, es probable en el fragmento multiplicativo de lógica lineal con exponencial !, y donde `nat` se interpreta como **1**.

43. Dar la derivación de tipos para el término $\lambda x.x + x$.