



UNIVERSIDAD NACIONAL DE ROSARIO
FACULTAD DE CIENCIAS EXACTAS, INGENIERÍA Y AGRIMENSURA
Licenciatura en Ciencias de la Computación
Análisis de Lenguajes de Programación

Intérprete de Cálculo Lambda Simple Tipado

Alumnos:

CRESPO, Lisandro (C-6165/4)
MISTA, Agustín (M-6105/1)

Docentes:

JASKELIOFF, Mauro
SIMICH, Eugenia
MANZINO, Cecilia
RABASEDAS, Juan Manuel

14 de Octubre de 2015

Ejercicio 1 Damos una derivación de tipo para el término S definido en *Prelude.lam* donde:

$$S = \lambda x:B \rightarrow B \rightarrow B . \lambda y:B \rightarrow B . \lambda z:B . (x \ z) \ (y \ z)$$

$$\frac{\frac{\frac{x:B \rightarrow B \rightarrow B \in \Gamma}{\Gamma \vdash x:B \rightarrow B \rightarrow B} T_{VAR} \quad \frac{\frac{z:B \in \Gamma}{\Gamma \vdash z:B} T_{VAR}}{\Gamma \vdash x \ z:B \rightarrow B} T_{APP} \quad \frac{\frac{\frac{y:B \rightarrow B \in \Gamma}{\Gamma \vdash y:B \rightarrow B} T_{VAR} \quad \frac{\frac{z:B \in \Gamma}{\Gamma \vdash z:B} T_{VAR}}{\Gamma \vdash y \ z:B} T_{APP}}{\Gamma \vdash y \ z:B} T_{APP}}{\frac{x:B \rightarrow B \rightarrow B, y:B \rightarrow B, z:B \vdash (x \ z) \ (y \ z):B}{x:B \rightarrow B \rightarrow B, y:B \rightarrow B \vdash \lambda z:B . (x \ z) \ (y \ z):B \rightarrow B} T_{ABS}} T_{APP} T_{ABS} T_{ABS}$$

$$\frac{\frac{x:B \rightarrow B \rightarrow B \vdash \lambda y:B \rightarrow B . \lambda z:B . (x \ z) \ (y \ z):(B \rightarrow B) \rightarrow (B \rightarrow B)}{\vdash \lambda x:B \rightarrow B \rightarrow B . \lambda y:B \rightarrow B . \lambda z:B . (x \ z) \ (y \ z):(B \rightarrow B \rightarrow B) \rightarrow (B \rightarrow B) \rightarrow B \rightarrow B} T_{ABS}} T_{ABS}$$

Por comodidad, llamamos: $\Gamma = x:B \rightarrow B \rightarrow B, y:B \rightarrow B, z:B$

Ejercicio 2

La función **infer** retorna un **Either String Type**, en lugar de un valor de tipo **Type**, ya que en caso de que la inferencia de tipo falle, retorna un **String** con detalles del error.

El operador $>>=$ nos permite pasar valores no monádicos a funciones sin salir de una mónada. En nuestro caso, $>>=$, toma un **Either** v y una función f y retorna **Left** v si es una cadena y $f \ v$ en otro caso. Esto es útil para la propagación de errores sin necesidad de hacer *pattern matching* sobre los resultados previos.

Ejercicio 5 Damos una derivación de tipo para el término:

$$(\text{let } z = ((\lambda x:B . x) \text{ as } B \rightarrow B) \text{ in } z) \text{ as } B \rightarrow B$$

$$\frac{\frac{\frac{\frac{x:B \in x:B}{x:B \vdash x:B} T_{VAR}}{\vdash \lambda x:B . x:B \rightarrow B} T_{ABS}}{\vdash (\lambda x:B . x) \text{ as } B \rightarrow B:B \rightarrow B} T_{ASCRIBE} \quad \frac{\frac{z:B \rightarrow B \in z:B \rightarrow B}{z:B \rightarrow B \vdash z:B \rightarrow B} T_{VAR}}{\vdash z:B \rightarrow B} T_{LET} T_{ASCRIBE}$$

$$\frac{\vdash (\text{let } z = ((\lambda x:B . x) \text{ as } B \rightarrow B) \text{ in } z):B \rightarrow B}{\vdash (\text{let } z = ((\lambda x:B . x) \text{ as } B \rightarrow B) \text{ in } z) \text{ as } B \rightarrow B:B \rightarrow B} T_{ASCRIBE}$$

