

# Lenguajes de Programación 2023-1

## Boletín de ejercicios 5

Javier Enriquez Mendoza

Kevin Axel Prestegui Ramos  
Oscar Fernando Millán Pimentel

Ramón Arenas Ayala

16 de noviembre de 2022

### Ejercicios. .

1. Obtenga el tipo de las siguientes expresiones mediante el algoritmo de inferencia de tipos calculando el conjunto de restricciones y ejecutando el algoritmo de unificación.

<code>lam x ⇒ let y = 2 in</code>	<code>lam n ⇒ lam m ⇒</code>
<code>    lam z ⇒ x + (z + y)</code>	<code>    if n = 0 then</code>
<code>end</code>	<code>    (if m = 0 then true else false)</code>
	<code>else m ≤ n</code>

`(recfun RGE n ⇒ lam m ⇒`  
`if n = 0 then`  
`(if m = 0 then true else false)`  
`else RLE (n - 1)(m - 1))`

2

2. Yo Lambda desea extender el lenguaje EAB con un nuevo operador `match` para definir expresiones mediante un análisis de casos sobre números (cero y sucesor).

- Sintaxis: `match e1 with 0 ⇒ e2 | suc x ⇒ e3 end`
- Semántica: Se evalúa  $e_1$  a un valor  $v$ , si  $v = 0$  se devuelve  $e_2$  y si  $v = \text{suc } x$  se devuelve  $e_3$ .

Realiza lo siguiente:

- a) Define la sintaxis abstracta para `match`. Nota que hay un ligado explícito en  $e_3$
- b) Define la semántica operacional para esta extensión.
- c) Define la semántica estática para esta extensión.
- d) Extiende el algoritmo de inferencia de tipos al agregar el caso de definición de restricciones para este operador
- e) ¿Cómo definirías el operador `match` mediante azúcar sintáctica?