

Fecha de entrega: 12 de octubre del 2018 (Examen parcial 2 ese mismo día).

(Si redactas por completo tu tarea en L<sup>A</sup>T<sub>E</sub>X tienes un punto extra)

Considera la gramática:

$e ::= x \mid n \mid \text{true} \mid \text{false} \mid e+e \mid \text{if } e \text{ then } e \text{ else } e \mid \text{iszero } e \mid \text{let } x = e \text{ in } e \text{ end} \mid e < e \mid e = e \mid \neg e$

con  $n \in \mathbb{Z}$ . La extensión al paradigma imperativo se hace de la siguiente manera:

$\ell_n \mid e_1 := e_2 \mid \text{ref } e \mid !e \mid e_1; e_2 \mid \text{while } e_1 \text{ do } e_2 \mid ()$

1. Escribe las reglas de tipado (semántica estática) para cada una de las expresiones anteriores.
2. Considera los siguientes programas:

■  $p_1 \Leftarrow$

```
let x = ref (iszero (3+4)) in
  let y = ref (if !x then 3 else 4) in
    let z = if !y < 10 then !y + 6 else 7+!y
    end
  end
```

■  $p_2 \Leftarrow$

```
let z = ref 5 in
  let w = ref 3 in
    while (0<!w)
      z := 5+3;
      w := !w-1
    end
  !z
end
```

■  $p_3 \Leftarrow$

```
let z = ref 10 in
  let w = ref 7 in
    while (0<!w)
      z := !z-1;
      w := !w-1
    end
  if !z = 3 then true else false
end
```

Escribe la derivación de tipos de los tres programas descritos antes.

3. Define listas ligadas como estructuras de datos efímeras y define una función que obtenga el último elemento de dichas listas (puedes combinar sintaxis de Haskell como se vio en clase).