



Primer examen opcional - Fundamentos de lenguajes de programación

Duración: 2 horas

Carlos Andres Delgado S, Ing *

07 de Mayo 2018

1. Abstracción de datos [50 puntos]

Dada la siguiente gramática:

```
<arbol-ternario> := <arbol-vacio>
                  := <int> <arbol-ternario>
                  := <arbol-ternario>
                  := <arbol-ternario>
```

Los constructores se llaman arbol-vacio y arbol-novacio respectivamente.

Se desea diseñar una **representación basada en procedimientos y en datatypes** para

- Obtener la suma de los elementos del árbol. Función observadora **sumaElementosArbol(arbol)**
- Obtener el mayor elemento del árbol. Función observadora **maximoArbol(arbol)**

1. (5 puntos) Declare un árbol con al menos 5 nodos no vacíos. Dibuje el árbol que está declarando
2. (25 puntos) Diseñe la representación basada en procedimientos: Especifique las funciones constructoras y observadoras en lenguaje Scheme.
3. (20 puntos) Diseñe la representación basada en Datatypes, especifique las funciones solicitadas en lenguaje Scheme.

2. Evaluación de expresiones

2.1. Paso por valor [25 puntos]

Dado el ambiente ambiente inicial *env0* con identificadores (**a b c f**) y valores (**1 2 3 (closure '(a b c) +(a,-(b,c)) empty-env)**)

```
let
  a = proc(x y) -(x,y)
  b = proc(x y) *(x,y)
  c = (f c a b)
in
  let
    g = proc(x y) if -(x,y) then (a x y)
                  else (b x y)
  a = (a c c)
  b = (b c c)
in
  letrec
    f (a, b) = if a then (g a b)
               else +(b, (f -(a,1) b))
  in
    (f +(a,c) +(a,b))
```

1. (5 puntos) Indique el valor de la expresión.

2. (20 puntos) Dibuje los ambientes que se generan y muestre mediante líneas de que ambientes extienden. Sea cuidadoso en la representación.

2.2. Paso por referencia [25 puntos]

Dado el ambiente ambiente inicial *env0* con identificadores (**x y z f**) y valores (**0 1 2 (closure '(a) begin set a=+(a,2); +(a,1) end empty-env)**)

```
let
  a = (f (f x))
  b = (f (f y))
in
  let
    g = proc(a b)
      begin
        set a = -(a,1);
        set b = +(b,z);
        set z = +(z,1);
        +(a,b)
      end
  in
    let
      s = (g (f a) (f b))
    in
      list(x,y,z,a,b,s)
```

1. (5 puntos) Indique el valor de la expresión.

2. (20 puntos) Dibuje los ambientes que se generan y muestre mediante líneas de que ambientes extienden. Sea cuidadoso en la representación.

Importante: En la expresión $s = (g(f a) (f b))$ primero se llama $(f a)$ y luego $(f b)$.

La primitiva `list`, retorna una lista con los datos que se le envían.

*carlos.andres.delgado@correounivalle.edu.co