



## Segundo examen parcial Fundamentos de lenguajes de programación

Duración: 2 horas

Carlos Andres Delgado S, Ing \*

19 de Febrero 2015

Nombre: \_\_\_\_\_  
Código: \_\_\_\_\_

■  $b = (f\ b\ c\ a);$   
 $x :$  \_\_\_\_\_  
 $y :$  \_\_\_\_\_  
 $z :$  \_\_\_\_\_

■  $c = (f\ c\ b\ a);$   
 $x :$  \_\_\_\_\_  
 $y :$  \_\_\_\_\_  
 $z :$  \_\_\_\_\_

■ valor de la expresión:  
\_\_\_\_\_

### 1. Paso de Parámetros por Referencia [12 pts.]

```
1 let
2   x = 3
3   y = 4
4   z = 8
5
6   m = proc(s j k)
7     begin
8       set s = +(j,k);
9       +(s,2)
10    end
11
12   n = proc(a b c f)
13     begin
14       set a = (f a b c);
15       set b = (f b c a);
16       set c = (f c b a);
17       +(a, *(b,c))
18     end
19
20 in
21   begin
22     set z = (n x y z m);
23     +(z, y)
24   end
```

Si el paso de parámetros es *por referencia*:

- a) (6 pts) Dibuje el ambiente en el cual se evalúa la expresión.
- b) (6 pts) Indique el valor de la expresión e indique los valores que toman  $x$ ,  $y$  y  $z$  en los llamados:

■  $a = (f\ a\ b\ c);$   
 $x :$  \_\_\_\_\_  
 $y :$  \_\_\_\_\_  
 $z :$  \_\_\_\_\_

### 2. Inferencia de Tipos [16 pts.]

En clase se describió el proceso formal de inferencia del tipo de una expresión y su implementación. En este ejercicio se pretende comprobar su asimilación del proceso (no de la implementación).

Para la expresión a continuación y las variables de tipo introducidas describa las ecuaciones de tipo generadas entre ellas. Acto seguido resuelva el sistema para encontrar el tipo de la expresión:

```
1 let
2   x = 3
3   y = 4
4
5   f1 = proc(a b c d)
6     if b then (a c d)
7     else proc(t w) *(t,w)
8
9   f2 = proc(n m)
10     proc(g h) +(*(g,m), *(h,n))
11
12   f3 = proc(i j) >(i,j)
13
14 in
15
16   ((f1 f2 (f3 x y) +(y,3) *(x,2)) x +(y,3))
```

es: \_\_\_\_\_

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

Expresión o Variable ligada	Variable de tipo
$f_1$	$t_{f_1}$
$f_2$	$t_{f_2}$
$f_3$	$t_{f_3}$
$x$	$t_x$
$y$	$t_y$
$a$	$t_a$
$b$	$t_b$
$c$	$t_c$
$d$	$t_d$
$t$	$t_t$
$w$	$t_w$
$n$	$t_n$
$m$	$t_m$
$g$	$t_g$
$h$	$t_h$
$i$	$t_i$
$j$	$t_j$
$(a \ c \ d)$	$t_1$
$\text{proc}(t \ w) \ *(t,w)$	$t_2$
$*(t,w)$	$t_3$
$\text{proc}(g \ h) \ +(*(g,m), *(h,n))$	$t_5$
$+(*(g,m), *(h,n))$	$t_6$
$*(g,m)$	$t_7$
$*(h,n)$	$t_8$
$>(i,j)$	$t_9$
$(f_1 \ f_2 \ (f_3 \ x \ y) \ +(y,3) \ *(x,2))$	$t_8$
$+(y,3)$	$t_9$
$*(x,2)$	$t_{10}$
$(f_3 \ x \ y)$	$t_{11}$
$((f_1 \ f_2 \ (f_3 \ x \ y) \ +(y,3) \ *(x,2)) \ x \ +(y,3))$	$t_{total}$

Expresión	Ecuacion(es) de tipo asociadas
$(f_3 \ x \ y)$	
$(f_1 \ f_2 \ (f_3 \ x \ y) \ +(y,3) \ *(x,2))$	
$((f_1 \ f_2 \ (f_3 \ x \ y) \ +(y,3) \ *(x,2)) \ x \ +(y,3))$	

Expresión	Ecuacion(es) de tipo asociadas
<pre>f1 = proc(a b c d)   if b then (a c d)   else proc(t w) *(t,w)</pre>	
<pre>f2 = proc(n m)   proc(g h)     +(* (g,m), *(h,n))</pre>	
<pre>f3 = proc(i j) &gt;(i,j)</pre>	

### 3. Claridad Operativa POO [16 pts.]

Considere el siguiente programa en nuestro lenguaje OO:

```

1 class c1 extends object
2   field a
3   field b
4
5   method initialize() 0
6
7   method setup(u,v)
8     begin
9       set a = +(u,v);
10      set b = *(u,3);
11      +(a,b)
12    end
13
14   method m1(x,y)
15     send self m2(+(x,b),-(y,a))
16
17   method m2(w,z)
18     +(w,+(z,a))
19
20 class c2 extends c1
21   field a
22   field b
23   field c
24
25   method initialize() 0
26
27   method setup(u,v)
28     begin
29       set a = -(u,3);
30       set b = +(v,3);
31       set c = +(u,v);
32       super setup(+(a,b),+(b,c))
33     end
34
35   method m1(x,y)
36     super m1(x,y)
37   method m2(w,z)
38     *(-(w,b),c),+(z,a))
39
40 let
41   o1 = new c1()
42   o2 = new c2()
43
44 in
45   let
46     x = send o1 setup(1,4)
47     y = send o2 setup(3,4)
48     i = send o1 m1(4,5)
49     j = send o1 m2(3,4)
50     k = send o2 m1(1,3)
51     l = send o2 m2(1,2)
52   in
53     +(+ (i,j), -(k,l))
```

a) [6 pts.] Muestre la representación de los objetos *o1* y *o2* después de la ejecución de:

- send o1 setup(1,4)
- send o2 setup(3,4)

Usted es libre de usar la representación de objetos simples o planos.

b) [10 pts.] Complete en la siguiente tabla, los valores asociados a las variables indicadas

en **después** de los momentos de evaluación señalados, **Importante:** En esta tabla sólo se le van a preguntar valores de campos de la clase o variables declaradas en el let, no se va a preguntar el estado de variables dentro de los métodos.

Variable	Valor	Al evaluar la expresión
$x$		$x = \text{send } o1 \text{ setup}(1,4)$
$y$		$y = \text{send } o2 \text{ setup}(3,4)$
$a$ de $c1$		$i = \text{send } o1 \text{ m1}(4,5)$
$b$ de $c1$		$i = \text{send } o1 \text{ m1}(4,5)$
$i$		$i = \text{send } o1 \text{ m1}(4,5)$
$a$ de $c1$		$j = \text{send } o1 \text{ m2}(3,4)$
$b$ de $c1$		$j = \text{send } o1 \text{ m2}(3,4)$
$j$		$j = \text{send } o1 \text{ m2}(3,4)$
$a$ de $c1$		$k = \text{send } o2 \text{ m1}(1,3)$
$b$ de $c1$		$k = \text{send } o2 \text{ m1}(1,3))$
$a$ de $c2$		$k = \text{send } o2 \text{ m1}(1,3)$
$b$ de $c2$		$k = \text{send } o2 \text{ m1}(1,3))$
$c$ de $c2$		$k = \text{send } o2 \text{ m1}(1,3)$
$k$		$k = \text{send } o2 \text{ m1}(1,3)$

Variable	Valor	Al evaluar la expresión
$a$ de $c1$		$l = \text{send } o2 \text{ m2}(1,2)$
$b$ de $c1$		$l = \text{send } o2 \text{ m2}(1,2)$
$a$ de $c2$		$l = \text{send } o2 \text{ m2}(1,2)$
$b$ de $c2$		$l = \text{send } o2 \text{ m2}(1,2)$
$c$ de $c1$		$l = \text{send } o2 \text{ m2}(1,2)$
$l$		$l = \text{send } o2 \text{ m2}(1,2)$
$+(+(i,j), -(k,l))$		$\text{in } +((i,j), -(k,l))$