

# Lenguajes de Programación

## Ejercicio Semanal 3

Sandra del Mar Soto Corderi  
Edgar Quiroz Castañeda

22 de agosto del 2019

1. Responde cada inciso a partir de la siguiente expresión  $e$

```
let x = let y = 3 in 2 + y end in
let w = let y = 2 in y * 5 end in
let y = w / x in y + x end + 2
end
end
```

- (a) Llenar la siguiente tabla con base en  $e$ . El orden de las expresiones let es, por supuesto, el orden de lectura como texto.

let	variable ligada	expresión a ligar	alcance del let
1	x		
2	y		
3	w		
4	y		
5	y		

- (b) Encuentra la representación en la sintaxis abstracta de orden superior de  $e$ . Está permitido usar sintaxis concreta para operaciones y números.
- (c) Encuentra una expresión  $e'$  que sea  $\alpha$ -equivalente a  $e$  y en donde todas las variables ligadas tengan distinto nombre.
- (d) ¿Cuál es el valor de  $e$ ? Explica como se llegó al resultado.

```

let x = let y = 3 in 2 + y end in
let w = let y = 2 in y * 5 end in
let y = w/x in y + x end + 2
end
end
→ (eletf)
let x = (2 + y)[y := 3] in
let w = (y * 5)[y := 2] in
let y = w/x in y + x end + 2
end
end
=
let x = (2 + 3) in
let w = (2 * 5) in
let y = w/x in y + x end + 2
end
end
→ (esumf), (eprodf), (eleti)
let x = 5 in
let w = 10 in
let y = w/x in y + x end + 2
end
end
→ (eletf)
(let y = w/x in y + x end + 2)[x := 5, w := 10]
=
let y = (w/x)[x := 5, w := 10] in (y + x)[x := 5, w := 10] end + 2
→ (eleti)
let y = 10/5 in y + 5 end + 2
→ (eprodf), (eleti)
let y = 2 in y + 5 end + 2
→ (eletf)
(y + 5)[y := 2] + 2
=
2 + 5 + 2
→ (esumf)
9

```

2. Realiza la siguiente sustitución mostrando la respuesta paso a paso:

$(\text{let } x = y * 4 \text{ in } (\text{let } z = x + 3 \text{ in } y * 2 \text{ end}) * y \text{ end}) [y := x + 2]$

Primero, tenemos que  $x \in FV(x + 2)$ , por lo que no se puede realizar una sustitución textual en el **let** exterior.

$(\text{let } x = y * 4 \text{ in } (\text{let } z = x + 3 \text{ in } y * 2 \text{ end}) * y \text{ end}) [y := x + 2]$

Entonces, hay que encontrar una expresión  $\alpha$ -equivalente que no tenga este problema.

$\equiv_{\alpha} (\text{let } w = y * 4 \text{ in } (\text{let } z = w + 3 \text{ in } y * 2 \text{ end}) * y \text{ end}) [y := x + 2]$

Luego, ya es posible realizar la sustitución de forma textual.

Primero, se aplica recursivamente la sustitución a la asignación y al cuerpo del **let** exterior.

$$= \text{let } w = (y * 4)[y := x + 2] \text{ in } ((\text{let } z = w + 3 \text{ in } y * 2 \text{ end}) * y)[y := x + 2] \text{ end}$$

En la expresión de asignación ya se puede aplicar directamente la sustitución. En el cuerpo del **let** aún hay que realizarla recursivamente.

Notemos que en el **let** interior, se puede realizar la sustitución textual sin necesidad de realizar ninguna  $\alpha$ -equivalencia.

$$= \text{let } w = (x + 2) * 4 \text{ in } (\text{let } z = w + 3 \text{ in } y * 2 \text{ end})[y := x + 2] * y[y := x + 2] \text{ end}$$

Se aplica la sustitución donde es posible, y se sigue recursivamente donde no se puede.

$$= \text{let } w = (x + 2) * 4 \text{ in } (\text{let } z = (w + 3)[y := x + 2] \text{ in } (y * 2)[y := x + 2] \text{ end}) * (x + 2) \text{ end}$$

Finalmente, se termina de aplicar la sustitución en todas las subexpresiones.

$$= \text{let } w = (x + 2) * 4 \text{ in } (\text{let } z = (w + 3) \text{ in } ((x + 2) * 2) \text{ end}) * (x + 2) \text{ end}$$