

Lenguajes de Programación, 2021-1

Tarea 5

Karla Ramírez Pulido

Manuel Soto Romero

Alejandro Hernández Mora

Alma Rocío Sánchez Salgado

Silvia Díaz Gómez

Fecha de inicio: 25 de noviembre de 2020

Fecha de entrega: 7 de diciembre de 2020

1. Evalúa la siguiente expresión usando el tipo de alcance y régimen de evaluación que se indica. Es necesario incluir el ambiente final en forma de pila en cada caso.

```
{with {a {+ 2 2}}
  {with {b {+ a a}}
    {with {foo {fun {x} {- x b}}}
      {with {a {- 2 2}}
        {with {b {- a a}}
          {foo -3}}}}}}
```

- (a) Alcance estático y evaluación glotona
- (b) Alcance dinámico y evaluación glotona
- (c) Alcance estático y evaluación perezosa
- (d) Alcance dinámico y evaluación perezosa

2. Dada la siguiente función:

```
(define (goo l)
  (if (empty? l)
      empty
      (append (car l) (goo (cdr l)))))
```

- (a) Explica qué hace y dale un nombre mnemotécnico.
- (b) Muestra los registros generados cuando es llamada con el argumento '((1 2) (3 4) (4 6)). ¿Cuántos registros son generados? ¿Cuántos son ocupados a la vez?
- (c) Optimiza la función usando la técnica de recursión de cola.
- (d) Muestra los registros generados por la función del inciso anterior con el argumento '((1 2) (3 4) (4 6)). ¿Cuántos registros son generados? ¿Cuántos son ocupados a la vez?

3. Dada la siguiente función:

```
(define (foo n)
  (if (< n 10)
      n
      (+ (modulo n 10) (foo (quotient n 10)))))
```

- (a) Explica qué hace y dale un nombre mnemotécnico.
- (b) Muestra los registros generados cuando es llamada con el argumento 1729. ¿Cuántos registros son generados? ¿Cuántos son ocupados a la vez?
- (c) Optimiza la función usando la técnica de recursión de cola.
- (d) Muestra los registros generados por la función del inciso anterior con el argumento 1729. ¿Cuántos registros son generados? ¿Cuántos son ocupados a la vez?

4. Dada la siguiente función:

```
(define (hoo n l)
  (if (zero? n)
      1
      (hoo (sub1 n) (cdr l))))
```

- (a) Explica qué hace y dale un nombre mnemotécnico.
- (b) Muestra los registros generados cuando es llamada con los argumento 3 y '(1 2 3 4). ¿Cuántos registros son generados? ¿Cuántos son ocupados a la vez?
- (c) Optimiza la función usando la técnica de recursión de cola.
- (d) Muestra los registros generados por la función del inciso anterior con los argumento 3 y '(1 2 3 4). ¿Cuántos registros son generados? ¿Cuántos son ocupados a la vez?

5. Evalúa la siguiente expresión usando el intérprete para cajas visto en clase. Debes usar alcance estático y evaluación glotona. Mostrar el ambiente (*stack*) y memoria (*heap*) finales.

```
{with {a {newbox 17}}
  {with {b {newbox 29}}
    {with {foo {fun {} {setbox a {openbox a}}}}
      {with {foo {fun {} {setbox b {openbox a}}}}
        {seqn {foo}
          {+ {openbox a} {openbox b}}}}}}}}
```

6. Dada la definición de la función `next-location` que genera nuevas direcciones de memoria, vista en clase, modifícala para que no tenga efectos secundarios, es decir, que no dependa de ninguna variable externa.

Hint: Modifícala usando la técnica *Store Passing Style* puedes añadir los parámetros que necesites.

```
(define last-location -1) ;; al inicio del intérprete

;; next-location: number
(define (next-location)
  (begin
    (set! last-location (add1 last-location))
    last-location))
```