

Lenguajes de Programación

Auxiliar N°5

Auxiliares: Kenji Maillard

17/04/2020

1 Ejercicios

1. *Ambientes con scope estatico y dinamico.*

```
(define (foo x y) (+ x y))

(define (f x)
  (let ([z 3]
        [h (let ([y 2]) (λ (u v) (foo u z)))])
    ((λ (z y) (h 5 y)) 8 9)))
(f 7)
```

En la ultima expresión `(f 7)`:

- (a) Describe el ambiente de la clausura `h`.
- (b) Describe el ambiente en el que `foo` está llamado.
- (c) Si Racket tenía scope dinamico, en que ambiente sería llamado `foo`.

2. *Clausuras.*

- (a) Escriba un caso que muestre que el interprete de FWAE puede capturar demasiados identificadores en las clausuras.
- (b) En teoria, ¿Puede ser problematico capturar demasiados identificadores en una clausura? ¿Y en practica?
- (c) Comente que cambios hay que hacer para arreglar el interprete para asegurar que las clausuras solamente capturan los identificadores necesarios.

3. *Orden de evaluacion de `if`.* Recuerden que se puede definir booleanos y condicional con funciones de orden superior:

```
(define (eager-true x y) x)
(define (eager-false x y) y)
(define (eager-if c l r) (c l r))
```

- (a) ¿Son equivalente el `if` y `eager-if`? Argumente.
 - (b) Utilizando funciones de primera clase, defina booleanos que tienen el mismo comportamiento que `if`.
4. *Streams*. Un stream es una lista perezosa donde los elementos no se evalúan de manera temprana, sino cuando es necesario.

```
(define s (stream 3 (error "Interrupt") 7 22))
(test (stream-first s) 3)
(test/exn (stream-ref s 2) "Interrupt")
```

Eso permite definir stream potencialmente infinito.

- (a) Defina una función `nats` que toma un número y retorna el stream de los números más grande.
- (b) Defina una función `eratosthenes` que implementa el criba de Eratóstenes. Se puede utilizar la función `divides?` que se encuentra en el módulo `math/number-theory` (debes añadir `(require math/number-theory)` para tener acceso a la función).