

# Lenguajes de Programación

## Auxiliar N°3

Auxiliar: Fabián Mosso

01/04/2016

Recuerde enunciar el contrato y tests.

```
(deftype Expr
  (num n)
  (add l r)
  (sub l r)
  (if0 c t f)
  (with id val body)
  (id x))
```

1. Dada la función de reemplazo

```
(define (subst expr sub-id val)
  (match expr
    [(num n) expr]
    [(add l r) (add (subst l sub-id val)
                     (subst r sub-id val))]
    [(sub l r) (sub (subst l sub-id val)
                    (subst r sub-id val))]
    [(if0 c t f) (if0 (subst c sub-id val)
                      (subst t sub-id val)
                      (subst f sub-id val))]
    [(with bound-id named-expr body)
     (if (symbol=? bound-id sub-id)
         expr
         (with bound-id named-expr
              (subst body sub-id val)))]
    [(id x) (if (symbol=? x sub-id) val expr)]))
```

Explique porque se cae en los siguientes casos y arregle la función:

- `{with {x 5} {with {y x} y}}`
- `{with {x 5} {with {x x} x}}`

2. ¿Es posible definir funciones recursivas en un lenguaje de funciones con ambientes? Argumente.
3. Considere la siguiente definición del ambiente con hash y mutación:

```
(define empty-env make-hash)

(define (extend-env sym val env)
  (hash-set! env sym val)
  env)

(define (env-lookup x env)
  (hash-ref! env x (thunk (error "no value found"))))
```

A que evalúa el siguiente ejemplo?:

```
'{with {x 1}
  {+ {with {x 3}
      x}
    x}}
```

4. ¿Por qué no hay operación deshacer en el repositorio de substitución diferida? No supone esto un riesgo de eliminar el scope estático si una substitución reemplaza a una anterior?
5. Ejemplos de scope dinamico