

## Guía de Trabajos Prácticos IV – Programación Funcional

1. Convierta las siguientes expresiones aritméticas en expresiones Scheme y evalúelas:

- $7 + (2 * -1/3) + -10.7$
- $(7/3 * 5/9) \div (5/8 - 2/3)$
- $1 + 3 \div (2 + 1 \div (5 + 1/2))$
- $1 \times -2 \times 3 \times -4 \times 5 \times -6 \times 7$

2. Determine el valor de las siguientes expresiones. Use el DrScheme para verificar su respuesta

- (cons 'car '+)
- (list 'esto '(es muy fácil))
- (cons 'pero '(se está complicando...))
- (cons '(y ahora no se que ) 'hizo)
- (quote (+ 7 2))
- (cons '+ '(10 3))
- (car '(+ 10 3))
- (cdr '(+ 10 3))
- cons
- (quote (cons (car (cdr (7 4)))) )
- (quote cons)
- (car (quote (quote cons)))
- (+ 2 3)
- (+ '2 '3)
- (+ (car '(2 3)) (car (cdr '(2 3))))
- ((car (list + - \* /)) 2 3)

3. (car (car '((a b) (c d)))) devuelve a. Determine que composición de car y cdr aplicados a ((a b) (c d)) devuelven b, c y d.

4. El comportamiento de `(car (car (car '((a b c) (d))))))` no está definido porque `(car '((a b c) (d)))` es `(a b c)`, `(car '(a b c))` es `a`, y `(car 'a)` no está definido.  
Determine todas las combinaciones legales de `car` y `cdr` aplicables a `((a b) (c d))`.
5. Describa los pasos necesarios para evaluar la siguiente expresión:  
`((car (cdr (cdr (list + - * /)))) 5 5)`
6. Obtenga el elemento `x` de las siguientes listas:
  1. `'( a b c . x)`
  2. `'(a b c x)`
  3. `'((a . x) b)`
  4. `'(x . a)`
  5. `'(a . x)`