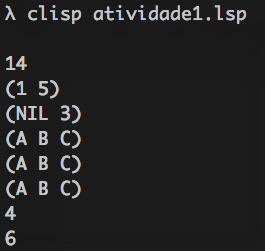
Nomes:

Natalia Cabrera Alves, 31210856

Clauber Pereira Stipkovic Halic, 31243045

Allan Gustavo Fernandes, 31568211

Lista 1 (Respostas e códigos):



**Códigos:**

; (a) - As expressões internas são executadas, soma e subitração,

; os resultados somados, retornando 14.

(print (+ (- 5 1) (+ 3 7)))

; (b) - Realiza a operação "+" com os argumentos e returna a lista (1 5).

(print (list 1 (+ 2 3)))

; (c) - se o objeto verificado com o listp for uma lista retorna T,

; mas no caso da comparação abaixo, ele retorna NIL, e não imprimi nada.

(if (listp 1)

(print ((+ 1 2) (+ 3 4)))

)

; (d) - Verifica se 3 é uma lista, retorna NIL, no "and" retorna NIL (pois T e NIL é NIL)

; e retorna a lista (NIL 3), pois também calculou a operação "+"

(print (list (and (listp 3) t) (+ 1 2)))

; (2)

(print (cons 'a (cons 'b (cons 'c nil))))

(print (cons 'a '(b c)))

(print (cons 'a (cons 'b '(c))))

; (3)

; Teste = (quartoElemento (list 1 2 3 4 5))

(defun quartoElemento(x)

(car (cdr (cdr (cdr x))))

)

(print (quartoElemento (list 1 2 3 4 5)))

; (4)

; Teste = (retornaMaior 5 6)

(defun retornaMaior(a b)

(if (> a b)

a

b

)

)

(print (retornaMaior 6 5))

; (5)

; (a)

(defun enigma (x)

(and (not (null x))

(or

(null (car x))

(enigma (cdr x))

)

)

)

; (b)

; TODO: entender isso, pq não faço idéia do que o enigma faz hauhau

(defun mystery (x y)

(if (null y)

nil

(if (eql (car y) x)

0

(let ((z (mystery x (cdr y))))

(and z (+ z 1))

)

)

)

)

; (6)

(defun elementoLista (x)

(if (null x)

NIL

(if (listp (car x))

t

(elementoLista (cdr x))

)

)

)

; ; (7)

; ; (a)

(defun imprimePontos (n)

(if (or (not (numberp n))

(<= n 0)

)

(format t "O argumento deve ser um inteiro positivo.~%")

(do ((i 1 (+ i 1)))

((> i n))

(format t ".")

)

)

)

;; (b)

(defun countElement (a L)

(cond

((null L) 0)

((equal a (car L)) (+ 1 (countElement a (cdr L))))

(t (countElement a (cdr L)))

)

)

;; (8)

(defun new-union (lst1 lst2)

(cond ((null lst1) lst2)

((null lst2) lst1)

(t (let ((elt (car lst1)))

(if (member elt lst2)

(cons elt (new-union (cdr lst1) (remove elt lst2)))

(cons elt (new-union (cdr lst1) lst2)))))))