

Lenguajes de Programación (CC41A) - Primavera 2009

Clase Auxiliar 2

Profesor: Tomás Barros

Auxiliar: Víctor Ramiro

1. Defina la siguiente función (`duple n x`) tal que repita *n-veces* el objeto *x*.

`duple :: Number, Symbol -> <List-of-Symbols>`

`(duple 5 3) -> (3 3 3 3 3)`

`(duple 2 '(ho ho)) -> ((ho ho) (ho ho))`

2. Defina la función (`filter-in p ls`) de tal manera que filtre los elementos de *ls* que no cumplan con el predicado *p*.

`filter-in :: (Symbol -> Boolean), <List-of-Symbols> -> <List-of-Symbols>`

3. Utilizando la función (`is-number? n`) que verifica si *n* es del tipo *Number*.

`is-number? :: Symbol -> Boolean`

Explique como se ejecuta y comente las implicancias de su uso.

`(filter-in is-number? '(1 () 2 (a b) c)) -> (1 2)`

4. Dada la definición de Árbol Binario(formato BNF).

`<BinTree> ::= ()`

`| (<Symbol> . <BinTree> . <BinTree>)`

defina el constructor de la especificación del tipo de dato *BinTree*

5. Escriba la siguiente API de acceso para manejar un árbol binario. Incluya el contrato correspondiente en su respuesta.

`(define (make-tree val left righth) ... crea un nodo de un árbol)`

`(define (value tree) ... valor de un nodo)`

`(define (left tree) ... sub-árbol izquierdo)`

`(define (righth tree) ... sub-árbol derecho)`

`(define (leave? tree) ... predicado que dice si el árbol es una hoja)`

6. Escriba las siguientes operaciones que funcionan con la estructura de árbol definida anteriormente. Incluya el contrato correspondiente en su respuesta.

(define (size? tree) ... retorna la altura de un árbol binario)

(define (sum-up tree) ... retorna la suma de los nodos del árbol)

7. Escriba la función que crea un *BinTree* apartir de una *List-of-Symbols*. La inserción se hace de foma ordenada.

`make-bintree :: <List-of-Symbols> -> BinTree`

8. Cree la función (`map-bintree f bt`) que aplica (`map`) la función f a todos los nodos del árbol.

`map-bintree :: (Symbol -> Symbol), BinTree -> BinTree`

9. **Propuesto:** Defina el tipo de Dato *NTree*, tal que es un árbol con una cantidad n de ramas con $n \geq 0$ y en sus nodos tienen datos del tipo *Symbol*. Luego para este nuevo tipo de dato redefina las funciones anteriores.