

PRÁCTICA PROGRAMACIÓN FUNCIONAL

Funciones estándar sobre listas

I. Utilizando las funciones estándar sobre listas, definir las siguientes funciones:

1. Definir una función que reciba una lista de elementos y devuelva el tercero
2. Definir una función que reciba una lista de elementos y devuelva el segundo
3. Definir una función que reciba una lista de listas de funciones y un elemento y aplique la 1ra función de la primera lista al elemento
4. Definir una función que reciba una lista de listas y devuelva el 5to. Elemento de la 3ra. lista.
5. Definir una función que reciba una lista de listas de listas y devuelva el 3er. elemento de la 4ta. Lista de la 2da. lista
6. Definir una función que verifique si una lista de listas podría ser considerada una matriz
7. Definir una función que reciba un número y una lista y devuelva el elemento de la lista que esta en la posición n
8. Definir las funciones length, filter, zipWith utilizando las otras funciones
9. Definir una función que compare 2 listas y devuelva True si las listas son iguales
10. Definir una función que fusione 2 listas ordenadas en una 3ra. ordenada (sin necesidad de ordenar).
11. Definir una función que verifique si una lista esta ordenada de acuerdo a una función de orden.
12. Definir una función que reciba una matriz y devuelva su transpuesta
13. Definir una función que reciba 2 matrices y las multiplique
14. Definir una función que reciba 3 matrices y las multiplique
15. Definir una función que reciba 4 matrices y las multiplique
16. Definir una función que reciba 1 matriz y una función de orden y devuelva True si la matriz esta ordenada de acuerdo a la función de orden.
17. Definir una función que reciba una lista de números y devuelva todos los números pares
18. Definir una función que reciba una lista de listas y devuelva una lista de sus longitudes.
19. Definir una función que reciba una lista de listas y devuelva solo aquellas cuya longitud sea par.
20. Definir una función que reciba una lista de listas de números y borre todos los números pares de estas listas
21. Definir una función que reciba una lista de listas y devuelva una lista formada por los penúltimos elementos de las listas
22. Definir una función que reciba un número y devuelva una lista con los posibles divisores del número.
23. Definir la función zipWith en terminos de zip

II. Reducir, si es posible, las siguientes expresiones

1. map (even.head) [[1,2,3,4],[6,7,1,2],[7,8]]
2. filter ((>2).head) [[1,2,3,4],[6,7,1,2],[7,8]]
3. takeWhile (even.head) [[1,2,3,4],[6,7,1,2],[7,8]]
4. dropWhile (even.head) [[1,2,3,4],[6,7,1,2],[7,8]]
5. ((\x,y) -> y) .last) (zip [1,2,3,4,5][map (*2), filter (>2)]) [1,2,6,8,3]
6. filter (even.length) (map (map (even.head.tail.tail)) [[[1,2,4,5,6],[8,3,5,2],[2,2,2]], [[1,2,4,5,6],[8,3,5,2]]])
7. map (\(x,y) -> x y) (zip [takeWhile even,filter odd,dropWhile (even.(+3))] [[2,7,8],[3,6,5,4],[2,5,7]])
8. map (map (head.tail.tail.init)) [[[2,7,8,1,3],[3,6,5,4,6,3],[2,5,7,1]], [[2,7,8,9,10],[1,2,3,6,5,4],[3,8,2,5,7]]]
9. (head.tail.tail) [(+5),(*6),((+3).(+4))] 10
10. (head.tail.tail) [(+),(*),((+3).(+4))] 10
11. map (map (map (*2))) [[[1,2,3],[4,5]],[[6,7]],[[8],[9,10]]]

III.

1. Sea:

$$(\text{map } f).(\text{map } g) \text{ xs} = \text{map } p \text{ xs}$$

De una definición apropiada para p

2. Sea:

$$(\text{filter } q) \bullet (\text{filter } p) = \text{filter } (p \oplus q)$$

de una definición apropiada para el operador \oplus