

Práctico 2: Introducción a la Programación Funcional

NOTA Los ejercicios con * son para resolver en su casa

1. Leer los capítulos 1 y 2 del libro Aprende Haskell por el bien de todos!.
2. Definir las siguientes funciones:

- `hd :: [a] -> a` retorna el primer elemento de una lista.
- `tl :: [a] -> [a]` retorna toda la lista menos el primer elemento.
- `last :: [a] -> a` retorna el último elemento de la lista.
- `init :: [a] -> [a]` retorna toda la lista menos el último elemento.

3. Defina una función máximo de tres, tal que `maxTres x y z` es el máximo valor entre `x`, `y`, `z`. Por ejemplo: `maxTres 6 7 4 = 7`.

4. Defina las siguientes operaciones sobre listas (vistas en el teórico): concatenar, tomar, tirar y `<`.

5. Defina una función `abs :: Int -> Int` que calcula el valor absoluto de un número.

6 *. Defina una función `edad :: (Nat,Nat,Nat) -> (Nat,Nat,Nat) -> Int` que dada dos fechas indica los años transcurridos entre ellas. Por ejemplo:

`edad (20,10,1968) (30,4,1987) = 18`

7. La disyunción excluyente `xor` de dos fórmulas se verifica si una es verdadera y la otra es falsa. Defina la función `xor` que calcule la disyunción excluyente a partir de la tabla de verdad.

*. Ahora defina la función `xor2` que calcule la disyunción excluyente pero sin que considere todos los posibles valores de las entradas. Cuál será la diferencia entre ambas definiciones?

8. Defina una función que dado un número natural, decida si el mismo es primo o no.

9 *. Defina una función que dado un número natural `n`, retorne la lista de todos los números naturales primos menores que `n`.

10. Defina una función que dada una lista, retorne la reversa de la misma.
11. Defina una lista de nmeros, devuelva la lista solo con los nmeros primos.
- 12 *. Defina una función que dada una lista decida si es un palíndromo o no.
13. Defina una función que dados tres números **a**, **b**, **c** devuelva la cantidad de raíces reales de la ecuación $ax^2 + bx + c = 0$