

Ejercicios Prácticos de Programación Declarativa

Sesión de laboratorio 3

Curso 2019/20

- Realizad los ejercicios de la Sesión 2 que no acabasteis y los dos de abajo, correspondientes a la Sesión 3.
 - Subid dos ficheros .hs al Campus Virtual antes de que acabe la clase. Uno con **todos** los ejercicios de la Sesión 2 (del 1 al 5) y otro con los dos ejercicios de la Sesión 3. Es suficiente con que los suba uno si lo hacéis entre dos.
 - Como siempre incluid comentarios, sin olvidar vuestros nombres en las primeras líneas del fichero.
1. Definid expresiones Haskell usando funciones de orden superior y/o listas intensionales para representar:
 - a) La lista $[1, -1, 2, -2, 3, -3, 4, -4, \dots]$.
 - b) Una lista infinita $[(0, 0), (0, 1), (1, 0), (0, 2), (1, 1), (2, 0), (0, 3), (1, 2), \dots]$, que sirva como enumeración de todas las parejas de números naturales.

Si se os ocurren varias maneras de hacerlo mejor.

2. Programa las siguientes funciones, usando orden superior y listas intensionales.
 - a) `sufijos xs` devuelve la lista de todos los sufijos de `xs`.
 - b) `sublists xs` devuelve la lista de todas las sublistas de `xs`.
 - c) `perms xs` devuelve la lista de todas las permutaciones de `xs`.
 - d) `sumandos n` devuelve la lista de todas las descomposiciones en sumandos positivos de `n`. Ejemplo: `sumandos 3 = [[1,1,1],[1,2],[3]]`.