

Programación Declarativa

Sesión de laboratorio 4 bis

Curso 2020/21

- Realiza los siguientes ejercicios individualmente en un mismo fichero `.hs`.
 - Escribe tu nombre al comienzo del fichero como líneas comentadas.
 - Incluye comentarios significativos y no olvides **declarar los tipos** de las expresiones que defines.
 - Sube el fichero al Campus Virtual antes de que acabe la clase.
1. Considera el tipo enumerado `Direccion` para representar movimientos y la función `destino` que definiste para resolver el ejercicio 1 de la Sesión de laboratorio 4. Define las siguientes funciones:
 - a) `trayectoria punto movs` que devuelva la lista de puntos por los que se pasa al aplicar la lista de movimientos `movs` al punto del plano `punto`.
 - b) `inferior n movs movs'` que devuelva `True` si la trayectoria determinada por `movs` a partir de cualquier `punto` del plano $n \times n$ nunca sube por encima de la determinada por `movs'`. Devuelve `False` en caso contrario.
 2. Define un tipo de datos polimórfico para representar árboles generales, en los que cada nodo tiene una información y n hijos ($n \geq 0$, y puede variar con cada nodo). No se consideran árboles vacíos.
 - a) Programa las siguientes funciones:
 - `listaHojas t`, que obtiene la lista de las informaciones de todas las hojas del árbol `t`.
 - `listaNodos t`, que obtiene la lista de las informaciones de todos los nodos del árbol `t`.
 - `repMax t`, que devuelve el árbol resultante de poner como información de todos los nodos del árbol `t` la información más grande que aparece en `t`.
 - b) Declara **explícitamente** el tipo de los árboles como instancia de la clase `Ord` (usando `instance`), de manera que el orden definido sea el mismo que resultaría de usar `deriving Ord`.
 - c) Declara el tipo de los árboles como instancia de la clase `Show`, de manera que la vista en pantalla de un árbol sea visualmente más atractiva que lo que nos da el poner simplemente `deriving Show`.