

20 de diciembre de 2011

Apellidos: Nombre:

INSTRUCCIONES

- Resuelve el examen en un archivo que tenga por nombre `laboratorio-DNI-NOMBRE.hs`, sustituyendo DNI por tu número de dni o pasaporte y NOMBRE por tus apellidos y nombre (separados por guiones).
- Escribe también lo siguiente en las primeras líneas de ese archivo: dni, apellidos y nombre, nombre del ordenador desde el que estás realizando el examen.

NOTA: en este examen es **obligatorio** especificar, de la forma más general posible, el tipo de las *funciones solicitada*. Para las demás funciones auxiliares que se definan no es necesario.

Ejercicio 1

En la orilla izquierda de un río se encuentran un cierto número de misioneros y de caníbales que pretenden cruzar a la orilla derecha. Para ello disponen de una barca con una capacidad limitada. El problema consiste en trasladar a todos los misioneros y caníbales teniendo en cuenta que nunca pueden quedar en una orilla un número de misioneros en menor cantidad que de caníbales.

Definir el tipo de datos `Orilla` como un sinónimo de una tupla de dos números de tipo **Int** y el tipo de datos `Barca` como un sinónimo de una cadena.

Definir un nuevo tipo de datos `MisionerosYCanibales` que tenga un único constructor con dos argumentos que sean del tipo `Orilla` y un argumento que sea del tipo `Barca`. Este tipo de datos representa un posible estado del problema, de tal forma que los dos primeros argumentos representan el número de misioneros y de caníbales (en ese orden) que hay en las orillas izquierda y derecha, respectivamente, del río y el tercer argumento representa la orilla ("Izquierda" o "Derecha") en la que se encuentra la barca. Siempre asumiremos que al construir un valor de este tipo de datos los argumentos proporcionados son correctos.

Definir la función `atravesarRio` que reciba un estado de `MisionerosYCanibales` y dos enteros de tipo **Int** (el número de misioneros y de caníbales que atraviesan el río) y devuelva el estado de `MisionerosYCanibales` resultante de hacer que los misioneros y los caníbales indicados atraviesen el río. Siempre asumiremos que todos los argumentos recibidos son correctos y que el movimiento se puede realizar.

Ejercicio 2

Consideremos la siguiente definición de un nuevo tipo de dato que representa de manera recursiva los polinomios con coeficientes enteros:

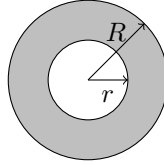
```
type Termino a = (a, a)
data Polinomio a = PolCero | Pol (Termino a) (Polinomio a)
deriving Show
```

Es decir, un polinomio con coeficientes enteros es el polinomio cero ($0x^0$) o un polinomio obtenido añadiendo un nuevo término con un cierto coeficiente entero y un cierto grado a un polinomio ya existente. Siempre asumiremos que no añadimos un término de grado igual a uno del polinomio ya existente, con la única excepción de que el término sea de grado 0, que se podrá añadir si el único término de grado 0 del polinomio ya existente es $0x^0$. Por otra parte, los términos no tienen por qué añadirse en orden creciente de grado.

Definir la función `coeficiente` que reciba un `Polinomio` y un número entero y devuelva el coeficiente del término del polinomio cuyo grado es el especificado. Si el polinomio no tiene ningún término con ese grado, entonces la función debe devolver el número 0. Siempre asumiremos que el polinomio recibido está construido de manera correcta.

Ejercicio 3

Una corona circular es una figura geométrica plana delimitada por dos circunferencias concéntricas.



Recordando que el área de un círculo de radio r se calcula mediante la fórmula πr^2 , definir una función que pida desde el teclado los radios de dos circunferencias concéntricas y escriba en pantalla el área de la corona circular que delimitan.