

PROGRAMACIÓN DECLARATIVA - Programación Funcional - Control 1 - 16.4.2009

1. Completar el siguiente fragmento de programa de forma que: las funciones sean *polimórficas* cuando sea posible, deben estar contempladas todas las entradas del origen, y toda llamada debe acabar siempre. Poner un ejemplo de llamada de `f ? False`, indicando el valor al que se reduce (no es necesario mostrar la traza). [3]

```
f :: ?
f x True = ?
f x _    = f r (n==m)
           where n:m:r = g x
-- pueden ser necesarias mas reglas para f (aquí o antes)

g :: ?
-- regla(s) de g
```

2. Implementar una función Haskell para comprobar si un entero contiene un determinado dígito. Usarlo para hacer otro programa que selecciona de una lista de enteros los que contienen un determinado dígito. Por ejemplo, si el dígito es 3 y la lista es [4, 3, 533, 45, 5665, 34], se obtiene la lista [3, 533, 34]. [3]

3. Elegir A o B: [4]

A. Implementar una función Haskell para pasar un entero a una base dada, que puede ser mayor que 10. La salida será en forma de *string*. Por ejemplo, dado el entero 692, si la base es 16 la salida es “2B4”, si la base es 2 la salida es “1010110100”, etc.

B. Representamos como $C_{m,n}$ el *número combinatorio* “*m* sobre *n*”, con $0 \leq n \leq m$. Se llama *fila m-ésima*, o *m-fila*, a la lista $[C_{m,0}, C_{m,1}, \dots, C_{m,m}]$. En las *m*-filas se cumple que $C_{m,0} = 1$, $C_{m,n} = 1$ si $m = n$, y $C_{m,n} = C_{m-1,n} + C_{m-1,n-1}$ en otros casos.

- Hacer un programa Haskell que calcula la *m*-fila. Debe ser recursivo; es decir, debe calcular la *m*-fila a partir de la (*m*-1)-fila (salvo los posibles casos base).
- Hacer un programa Haskell que calcula $C_{m,n}$ usando la función anterior.
- Hacer un programa Haskell que, dados $C_{m,n}$ y *m*, obtiene *n* (si existe).