

PUNTUACIONES:	1	2	3	4	5	6	total
	1.5	1.5	1.5	1	1	3.5	10.0

si } deseo que se publique mi calificación si fuera negativa  
 no }

**1** Da el tipo y completa la siguiente definición del operador lógico implicación, a través de patrones

*infixl 5 ==>*

*(==>) :: ...*

*False ==> \_ = ...*

*True ==> ... = ...*

**2** Completar las siguientes definiciones:

*esMúltiploDe :: Int → Int → Bool*

*p 'esMúltiploDe' q = ...*

*type Año = Int*

*esBisiesto :: Año → Bool*

— un año es bisiesto si es múltiplo de 4, y siéndolo de 100, lo es también de 400.

— Por ejemplo, 1100 y 1103 no son bisiestos, pero 2000 y 1104 si lo son.

*esBisiesto a = a 'esMúltiploDe' 4 && ...*

¿Es bisiesto el año 0 con tal definición?

**3** Complete la siguiente definición, sabiendo que *menorQueSatisface p xs* devuelve el primer elemento de la lista *xs* que satisface el predicado *p*

*menorQueSatisface :: ...*

*menorQueSatisface p (x : xs) | not(p x) = menorQueSatisface ...*  
*| otherwise = ...*

**4** ¿ Que valor computa la expresión *menorQueSatisface esBisiesto [1000 ..]* ?

**5** Escribe la siguiente función para comprobar si un número natural representado con un *Integer* es impar:

*impar 0 = False*

*impar (n + 1) = ...*

**6** Completa la siguiente función sabiendo que debe calcular el segundo menor de una tupla de 5 valores

*segundoMenor :: (Integer, Integer, Integer, Integer, Integer) → Integer*

*segundoMenor (x, y, z, t, u) | x > y = segundoMenor (y, x, z, t, u)*  
*| y > z = segundoMenor (x, z, y, t, u)*  
*| y > t = segundoMenor (x, t, z, y, u)*  
*| y > u = ...*  
*| otherwise = ...*