

PUNTUACIONES:	1	2	3	4	total
	2.0	4.0	2.0	2.0	10.0

1 Completa la siguiente función sabiendo que debe calcular la mediana de una tupla de 5 valores

$$\text{mediana} :: (\text{Integer}, \text{Integer}, \text{Integer}, \text{Integer}, \text{Integer}) \rightarrow \text{Integer}$$

$$\text{mediana}(x, y, z, t, u) \mid x > z = \text{mediana}(z, y, x, t, u)$$

$$\mid y > z = \dots$$

$$\mid z > t = \text{mediana}(x, y, t, z, u)$$

...

y da un razonamiento basado en conjuntos bien construidos para probar su corrección.

2 Consideremos las siguientes declaraciones para representar y manipular los números naturales:

data $N = O \mid S N$ **deriving** *Show*

$_1 = S O; _2 = S _1; _3 = S _2; _4 = S _3$

instance *Num* N **where**

$y + O = y$	$\text{itera } g z O = z$
$y + S x = S(y + x)$	$\text{itera } g z (S n) = g n (\text{itera } g z n)$
$y * O = O$	
$y * S x = y * x + y$	

A Demuestra que el tipo de *itera* es $\text{itera} :: \dots$

En efecto:

B Sabiendo que n está en forma normal (FN), calcula las FNs de las expresiones

$$n + _1$$

$$\text{itera}(\text{const } S) n _1$$

C Demuestra por inducción que se verifica $\forall n . n :: N . _1 * n = n$

D Define la función *par* :: $N \rightarrow \text{Bool}$ y la suma (+) directamente a través de *itera*:

$$\text{par} = \text{itera} \dots$$

$$(+) = \text{itera} \dots$$

3] Pretendemos escribir el algoritmo *quicksort con dos pivotes*:

- 1.- tomar los dos primeros elementos de la lista en el orden $x \leq y$.
- 2.- *repartir* el resto de la lista en tres segmentos m_1 , m_2 y m_3 , donde m_1 contenga los menores que x , m_2 los comprendidos entre x e y (excluido y), y finalmente m_3 los restantes.
- 3.- ordenar por separado cada segmento y "juntarlos" de forma que resulte la lista original pero ordenada.

A] Describa una función para repartir una lista según dos valores x y y , con $x \leq y$:

Main > repartir 3 7 [1,5,3,9,2,7]

([1,2], [5,3], [9,7])

repartir :: Ord a => a → a → [a] → ([a], [a], [a])

repartir x y [] = ([], [], [])

repartir x y (u : us) ...

B] Describa el algoritmo *quicksort* que ordena según el esquema anterior:

quicksort :: Ord a => [a] → [a]

...

4] Se considera la sucesión $\{f_n\}_{n \geq 0}$ definida en forma inductiva en la forma siguiente:

$$f_0 = 1, \quad f_1 = -1, \quad \text{y para } n \geq 0, f_{n+2} = (n+1)f_n + nf_{n+1}$$

A] Describa una red de procesos que tenga como salida la lista infinita $[f_0, f_1, f_2, \dots]$. Escriba así mismo la ecuación HASKELL correspondiente

B] Use la red anterior para comprobar que la salida es la lista cíclica $[1, -1, 1, -1, 1, -1, \dots]$