

Programación Funcional

Prueba Escrita (Segunda Instancia) - 2016

Nombre:

CI:

1. Dada la siguiente definición:

$$\text{mapShow } x \ y \ z = [\text{show } x, \text{show } y \ ++ \ \text{show } z]$$

¿Cuál de las siguientes afirmaciones es correcta?

- (a) La función no compila correctamente
- (b) La función tiene tipo $\text{mapShow} :: (\text{Show } a, \text{Show } b, \text{Show } c) \Rightarrow a \rightarrow b \rightarrow c \rightarrow [\text{String}]$
- (c) La función tiene tipo $\text{mapShow} :: \text{Show } a \Rightarrow a \rightarrow a \rightarrow a \rightarrow [\text{String}]$
- (d) La función tiene tipo $\text{mapShow} :: (\text{Show } a, \text{Show } b) \rightarrow a \rightarrow b \rightarrow b \rightarrow [\text{String}]$

Respuesta: b)

2. Implemente como *foldr* una función

$$\text{minimo} :: \text{Ord } a \Rightarrow [a] \rightarrow (a, \text{Int})$$

que dada una lista no vacía, retorne un par con el menor elemento y la cantidad de veces que ocurre en la lista.

$$\begin{aligned} \text{minimo } (x : xs) &= \text{foldr } \text{step } (x, 1) \ xs \\ \textbf{where } \text{step } y \ (m, n) &| \ m == y \quad = (m, n + 1) \\ &| \ m > y \quad = (y, 1) \\ &| \ \text{otherwise} \quad = (m, n) \end{aligned}$$

3. Dada la siguiente definición:

$$\text{mucho} = \text{take } 5 \circ \text{dropWhile } (>0) \circ \text{map } (\lambda f \rightarrow f \ 0) \$ \text{iterate } ((+1) \circ) \ \text{id}$$

¿Cuál de las siguientes afirmaciones es correcta?

- (a) Al evaluarse *mucho* retorna [1, 2, 3, 4, 5]
- (b) Al evaluarse *mucho* diverge
- (c) Al evaluarse *mucho* retorna [0, 1, 2, 3, 4]
- (d) Al evaluarse *mucho* retorna [6..]

Respuesta: c)

4. Dada la siguiente definición:

$$pp = foldl (\lambda r \rightarrow const (r + 1)) 0$$

¿Cuál de las siguientes afirmaciones no es correcta?

- (a) $pp \equiv length$
- (b) La función no compila correctamente
- (c) $pp \equiv sum \circ map (const 1)$
- (d) $pp \equiv foldr (\backslash_r \rightarrow r + 1) 0$

Respuesta: b)

5. Dadas las siguientes definiciones:

```

data Tree = Empty | Node Tree Tree
data Nat = Z | S Nat
instance Eq Nat where
  Z == Z = True
  S n == S m = n == m
  _ == _ = False
eqzz t t' = zig t == zag t'
zig Empty = Z
zig (Node l r) = S (zag l)
zag Empty = Z
zag (Node l r) = S (zig r)
t1 = Node (Node Empty Empty) (Node t1 Empty)
t2 = Node t2 (Node t2 Empty)
t3 = Node t3 Empty

```

¿Cuál de las siguientes afirmaciones es correcta?

- (a) $eqzz t3 t1$ diverge
- (b) $eqzz t1 t2$ diverge
- (c) $eqzz t2 t2$ diverge
- (d) $eqzz t1 t1$ diverge

Respuesta: c)

6. Dada la siguiente definición:

$$\text{compone } f \ g \ x \ y = (\text{curry } g \circ \text{uncurry } f) (x, \text{True}) \ y$$

¿Cuál de las siguientes afirmaciones no es correcta?

- (a) La aplicación $(\text{compone } \text{const } \text{snd } 1 \ 2)$ retorna 2
- (b) La función tiene tipo $(a \rightarrow \text{Bool} \rightarrow b) \rightarrow ((b, c) \rightarrow d) \rightarrow a \rightarrow c \rightarrow d$
- (c) La aplicación $(\text{compone } (\text{curry } (\text{uncurry } (\wedge)))) \text{fst } \text{False } 2)$ retorna *False*
- (d) La aplicación $(\text{compone } (\text{curry } \text{snd}) \text{fst } 3 \ (\text{False}, 5))$ no compila correctamente

Respuesta: d)

7. Dadas las siguientes definiciones:

$$\begin{aligned} \text{twice } f &= f \circ f \\ \text{inf} &= 1 + \text{inf} \\ f &:: \text{Int} \rightarrow \text{Bool} \\ f \ x &= ((\text{twice } \text{twice } (+2)) \ x - \text{curry } \text{snd } \text{inf } 7 == \text{fst } (4, \text{inf})) \wedge (x > \text{inf}) \end{aligned}$$

Decir cual respuesta es válida:

- (a) $f \ x$ diverge para todo x
- (b) $f \ 4$ evalúa a *False*
- (c) $f \ 3$ evalúa a *False*
- (d) La función no compila correctamente

Respuesta: b)

8. Dadas las siguientes definiciones:

$$\begin{aligned} \text{data } \text{Arbol } a &= \text{Vacío} \mid \text{Nodo } (\text{Arbol } a) \ a \ (\text{Arbol } a) \\ \text{arma } [] &= \text{Vacío} \\ \text{arma } (x : xs) &= \text{Nodo } (\text{arma } rs) \ x \ (\text{arma } ls) \\ &\quad \text{where } (ls, rs) = \text{splitAt } (\text{length } xs \div 2) \ xs \\ \text{recorre } \text{Vacío} &= [] \\ \text{recorre } (\text{Nodo } l \ x \ r) &= \text{recorre } r \ ++ [x] \ ++ \text{recorre } l \end{aligned}$$

¿Cuál es el resultado de evaluar $(\text{recorre} \circ \text{arma}) [1, 2, 3, 4, 5]$?

- (a) [2, 3, 1, 4, 5]

- (b) [1, 3, 4, 5, 2]
- (c) [1, 2, 3, 4, 5]
- (d) [3, 1, 2, 5, 4]

Respuesta: a)

9. Dadas las siguientes definiciones:

$$\begin{aligned}
 a &= (fst\ a, b) \\
 b &= (repeat\ 5, c) \\
 c &= fst\ b
 \end{aligned}$$

Para cada una de las siguientes expresiones indique el resultado de su evaluación o si la misma diverge o si no compila.

- (a) $(fst\ a)$

diverge
- (b) $((take\ 8\ \circ\ snd)\ a)$

no compila
- (c) $((take\ 8\ \circ\ fst\ \circ\ snd)\ a)$

[5,5,5,5,5,5,5,5]
- (d) $((take\ 8\ \circ\ snd\ \circ\ snd)\ a)$

[5,5,5,5,5,5,5,5]
- (e) $(head\ \circ\ snd)\ b)$

5
- (f) $(length\ (foldr\ (\cdot)\ c\ [1, 2, 3, 4]))$

diverge
- (g) $(length\ (foldr\ (\cdot)\ [c]\ [[1], [2], [3], [4]]))$

5
- (h) $((head\ \circ\ map\ (*2)\ \circ\ snd\ \circ\ snd)\ a)$

10

10. Dada la siguiente definición:

$$mon\ a\ b = a \gg= \lambda x \rightarrow return\ b \gg= \lambda y \rightarrow return\ (x : y)$$

¿Cuál de las siguientes afirmaciones es correcta?

- (a) El tipo de mon es $mon :: Monad\ m \Rightarrow m\ a \rightarrow m\ [a] \rightarrow m\ [a]$
- (b) $mon\ 'a'\ ['b']$ retorna $['a', 'b']$
- (c) La función no compila correctamente
- (d) El tipo de mon es $mon :: Monad\ m \Rightarrow m\ a \rightarrow [a] \rightarrow m\ [a]$

Respuesta: d)