

Matemática I – Segunda Prueba Parcial – 29 de octubre de 2010

Ejercicio 1 (1,5 puntos)

Sean $f: A \rightarrow B$, $g: A \rightarrow (A \rightarrow B) \rightarrow B$, $h: (A \rightarrow A) \rightarrow A \rightarrow A$, $j: A \rightarrow A$, $p: (A \rightarrow B) \rightarrow B$, $x: A$, $y: B$
Indicar si las siguientes expresiones tienen tipo, en caso afirmativo indicarlo, justificando en todo caso la respuesta.

- a) $g \ x$ b) $g \ x \ x$ c) $g \ x \ (f \ x)$ d) $g \ x \ f$ e) $p \ f$ f) $p \ (g \ x)$

Ejercicio 2 (2,5 puntos)

- a) Definir en Haskell una función recursiva **sumcuad** que recibe un entero $x \geq 1$ y devuelve la suma de los cuadrados de los enteros desde 1 hasta x . Indicar su tipo.
Ejemplo: `sumcuad 5 = 56` (observar que $1+4+9+16+25 = 56$).
b) Dar la secuencia de cómputo para (`sumcuad 4`).

Ejercicio 3 (2,5 puntos)

- a) Definir por casos la función **sumaplust** que recibe dos naturales (\mathbb{N}) y devuelve el número natural siguiente a la suma de ambos (Indicar su tipo).
Ejemplo: `sumaplust (S(S Z)) (S Z) = S(S(S(S Z)))`
b) Dar la secuencia de cómputo para (`sumaplust (S(S Z)) (S(S Z))`).

Ejercicio 4 (2,5 puntos)

- a) Definir una función **divlista** cuyo tipo sea el siguiente: $\text{Integer} \rightarrow [a] \rightarrow ([a], [a])$, que dados un entero positivo y una lista, divida la lista en el elemento n -ésimo.
Ejemplo: `divlista 3 [-4,5,8,-3,7,0,1] = ([-4,5,8], [-3,7,0,1])`
b) Dar la secuencia de cómputo para (`divlista 2 [-5,0,1,3,7]`)

Ejercicio 5 (3 puntos)

- a) En Haskell definimos una función **matuno** de la siguiente forma, :

```
matuno :: [Integer] -> [Integer] -> [Integer]
matuno [] [] = []
matuno (a:x) [] = (a:x)
matuno [] (b:y) = (b:y)
matuno (a:x) (b:y) | (a <= b) = a:(matuno x (b:y))
                   | otherwise = b:(matuno (a:x) y)
```

¿Cuál es el comportamiento dicha función?

- b) ¿Cuál es el resultado si en la línea de comandos digitamos `matuno [-7,-3,2] [-10,1,8]`?. Detalle la secuencia de cálculos para esta entrada.
c) Probar por inducción que: $\forall l, m \in \text{List } (A)$, se cumple:
`largo(matuno l m) = largo l + largo m`