

Examen Parcial Introducción a los Algoritmos - 18 de Abril de 2016
Comisiones Turno Tarde

nota

1	2	3	4	5
---	---	---	---	---

Apellido y Nombre:

Cantidad de hojas entregadas: ___ (Numerar cada hoja.)

1. [10 pto(s)] Definir la función $dosIguales : (A, A, A) \rightarrow Bool$ que dada una tripla de elementos (de un tipo con igualdad) devuelve $True$ si al menos dos de ellos son iguales y $False$ si no. Ejemplos:

(I) $dosIguales.([], [], []) = True$

(II) $dosIguales.(1, 2, 3) = False$

2. (a) [15 pto(s)] Definir la función recursiva $sumaPares : [(Num, Num)] \rightarrow Num$, que dada una lista de pares de números retorna la lista resultante de haberlos sumado. Ejemplos:

(I) $sumaPares.[(3, 4), (1, 13), (2, 2)] = [7, 14, 4]$

(b) [5 pto(s)] Evaluar manualmente la función utilizando el ejemplo (I). Justificar cada paso.

3. (a) [15 pto(s)] Definir la función recursiva $cuantos : Num \rightarrow [Num] \rightarrow Num$ que dado un número n y una lista de números l retorna cuantas veces aparece n en l . Ejemplos:

(I) $cuantos.1.[2, 3, 4] = 0$

(II) $cuantos.1.[1, 1, 2, 1] = 3$

(b) [5 pto(s)] Usar la función anterior para definir la función $mas0que1 : [Num] \rightarrow Bool$ que retorna $True$ si hay más 0s que 1s en la lista, y $False$ si no. Ejemplos:

(I) $mas0que1.[0] = True$

(II) $mas0que1.[1, 1, 0] = False$

4. [20 pto(s)] Dadas las siguientes funciones

$$\begin{aligned} reverse.[] &= [] & \#.[] &= 0 \\ reverse.(x \triangleright xs) &= (reverse.xs) \triangleleft x & \#.(x \triangleright xs) &= 1 + \#xs \end{aligned}$$

$$\begin{aligned} [] \triangleleft x &= x \triangleright [] \\ (x \triangleright xs) \triangleleft y &= x \triangleright (xs \triangleleft y) \end{aligned}$$

demuestre por inducción la siguiente propiedad

$$\#(reverse.xs) = \#.xs$$

5. [30 pto(s)] Dadas las siguientes funciones

$$\begin{aligned} prod.[] &= 1 & saca1.[] &= [] \\ prod.(x \triangleright xs) &= x \star prod.xs & saca1.(x \triangleright xs) &= (x = 1 \rightarrow saca1.xs \\ & & & \square x \neq 1 \rightarrow x \triangleright saca1.xs) \end{aligned}$$

demuestre por inducción la siguiente propiedad

$$prod.saca1.xs = prod.xs$$