

MODELOS AVANZADOS DE COMPUTACIÓN

PRÁCTICA 2: FUNCIONES EN HASKELL

EJERCICIO 1: Sea la función `nsobrek` tal que `nsobrek n k` es el número de combinaciones de n elementos tomados de k en k ; es decir:

$$\binom{n}{k} = \frac{n!}{k! (n - k)!}$$

Se pide: definir la función en **haskell** en el mayor número de variantes que hemos visto en clase.

EJERCICIO 2: Definir la función `raíces` tal que `raíces a b c` es la lista de las raíces de la ecuación $ax^2 + bx + c = 0$.

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Se pide: definir la función en **haskell** en el mayor número de variantes que hemos visto en clase.

EJERCICIO 3: Sucesión de Fibonacci

Los números de Fibonacci quedan definidos por las ecuaciones

Los números de Fibonacci quedan definidos por las ecuaciones

$$f_0 = 0$$

$$f_1 = 1$$

$$f_n = f_{n-1} + f_{n-2}$$

Esto produce los siguientes números:

- $f_2 = 1$
- $f_3 = 2$
- $f_4 = 3$
- $f_5 = 5$
- $f_6 = 8$
- $f_7 = 13$
- $f_8 = 21$

y así sucesivamente.

Se pide: definir la función en **haskell** en el mayor número de variantes que hemos visto en clase.

EJERCICIO 4: Comprobar la pertenencia a una lista usando una función recursiva.

Pertenece a [b]

Ejemplos:

Pertenece 3 [2,3,5] == True

Pertenece 4 [2,3,5] == False

Se pide: definir la función en **haskell** en el mayor número de variantes que hemos visto en clase.

EJERCICIO 5: Seleccionar 2 retos del proyecto de Euler que se puedan resolver con lo visto hasta el momento e implementarlo con el mayor número de definiciones posibles.

<https://projecteuler.net/>

ENTREGA y MEMORIA

La entrega se realizará a través de la plataforma en un documento .ZIP, el cual debe contener:

- Un fichero ***ejercicioN.hs*** con las diferentes implementaciones por cada ejercicio
- Un documento APELLIDO1-APELLIDO2-NOMBRE_P2_Memoria.pdf que deberá contener lo siguiente:
 - Portada
 - Índice
 - Código fuente de cada implementación
 - Descripción de cada código.

FECHA LÍMITE DE ENTREGA

Miércoles 14 de Noviembre a las 23:59