

# Ejercicios Prácticos de Programación Declarativa

## Sesión de laboratorio 1

Curso 2021/22

- Realizad los siguientes ejercicios en un mismo fichero `.hs`.
- **Escribid vuestro nombre al comienzo del fichero como líneas comentadas.**
- Incluid comentarios explicativos.
- Es necesario declarar el tipo de las funciones Haskell que se definan.
- Subid el fichero al Campus Virtual antes de que acabe la clase. Solo lo tiene que subir uno si lo hacéis entre dos.

1. Define expresiones Haskell para los siguientes apartados:
  - a) Calcular cuántos años enteros, días restantes enteras, horas restantes enteras, minutos restantes enteros y segundos restantes hay en  $10^6$  segundos. Utiliza una tupla de manera que cada componente corresponda a uno de los resultados pedidos.
  - b) Generaliza el apartado anterior, convirtiendo el número de segundos a convertir en argumento de una función.
2. Define una función que dado un año devuelva **True** si es bisiesto y **False** en caso contrario.
  - a) Usando expresiones `if`.
  - b) Usando guardas.
3. Programa la función `media xs` que calcula la media aritmética de los elementos de una lista de números `xs`. Usa la función `length` que calcula el número de elementos de una lista. ¿Surge algún problema con los tipos? Utiliza la función de conversión `fromIntegral` para resolverlo.
4. Programa las siguientes funciones:
  - a) `num_digitos x` = número de dígitos del número entero `x`.
  - b) `reduccion x` = resultado del proceso de sumar los dígitos del entero `x`, sumar los dígitos del resultado obtenido, y así sucesivamente hasta obtener un número menor que 10. La reducción de un entero negativo es la de su valor absoluto.
5. Define la disyunción booleana por ajuste de patrones, pero de tres formas diferentes, cambiando el número de ecuaciones, o las combinaciones de patrones **True**, **False**, **x**, ... en cada ecuación, o el orden de ecuaciones, etc. y de manera que haya una versión estricta solo en el primer argumento, otra estricta solo en el segundo y otra estricta en los dos. Para que coexistan todas las definiciones en el mismo programa, puedes darles nombres (o usar operadores) diferentes.