

Funcional 1: Primeros ejercicios

Notas previas

En algunos ejercicios se van a utilizar algunas de las funciones que están en el Prelude por ej:

<code>mod 20 3 = 2</code>	el resto de la división entre 20 y 3 es 2.
<code>div 14 3 = 4</code>	parte entera de la división entre 14 y 3 es 4.
<code>max 8 10 = 10</code>	devuelve el max entre 2 números.
<code>min 9 15 = 9</code>	devuelve el min entre 2 números.

Ahora sí, los ejercicios...

Básicos

1. Definir la función **esMultiploDeTres/1**, que devuelve True si un número es múltiplo de 3, p.ej:

```
Main> esMultiploDeTres 9
True
```

2. Definir la función **esMultiploDe/2**, que devuelve True si el segundo es múltiplo del primero, p.ej.

```
Main> esMultiploDe 3 12
True
```

3. Definir la función **cubo/1**, devuelve el cubo de un número.
4. Definir la función **area/2**, devuelve el área de un rectángulo a partir de su base y su altura.
5. Definir la función **esBisiesto/1**, indica si un año es bisiesto. (Un año es bisiesto si es divisible por 400 o es divisible por 4 pero no es divisible por 100) Nota: Resolverlo reutilizando la función `esMultiploDe/2`
6. Definir la función **celsiusToFahr/1**, pasa una temperatura en grados Celsius a grados Fahrenheit.
7. Definir la función **fahrToCelsius/1**, la inversa de la anterior.
8. Definir la función **haceFrioF/1**, indica si una temperatura expresada en grados Fahrenheit es fría. Decimos que hace frío si la temperatura es menor a 8 grados Celsius.
9. Definir la función **mcm/2** que devuelva el mínimo común múltiplo entre dos números, de acuerdo a esta fórmula.

$$m.c.m.(a, b) = \{a * b\} / \{m.c.d.(a, b)\}$$

[Más información.](#)

Nota: Se puede utilizar gcd.

10. Dispersión

Trabajamos con tres números que imaginamos como el nivel del río Paraná a la altura de Corrientes medido en tres días consecutivos; cada medición es un entero que representa una cantidad de cm.

P.ej. medí los días 1, 2 y 3, las mediciones son: 322 cm, 283 cm, y 294 cm.

A partir de estos tres números, podemos obtener algunas conclusiones.

Definir estas funciones:

- dispersion**, que toma los tres valores y devuelve la diferencia entre el más alto y el más bajo. Ayuda: extender max y min a tres argumentos, usando las versiones de dos elementos. De esa forma se puede definir dispersión sin escribir ninguna guarda (las guardas están en max y min, que estamos usando).
- diasParejos**, **diasLocos** y **diasNormales** reciben los valores de los tres días. Se dice que son días parejos si la dispersión es chica, que son días locos si la dispersión es grande, y que son días normales si no son ni parejos ni locos. Una dispersión se considera chica si es de menos de 30 cm, y grande si es de más de un metro.

Nota: Definir **diasNormales** a partir de las otras dos, no volver a hacer las cuentas.

11. En una plantación de pinos, de cada árbol se conoce la altura expresada en cm. El peso de un pino se puede calcular a partir de la altura así: 3 kg x cm hasta 3 metros, 2 kg x cm arriba de los 3 metros. P.ej. 2 metros \Rightarrow 600 kg, 5 metros \Rightarrow 1300 kg. Los pinos se usan para llevarlos a una fábrica de muebles, a la que le sirven árboles de entre 400 y 1000 kilos, un pino fuera de este rango no le sirve a la fábrica. Para esta situación:

- Definir la función **pesoPino**, recibe la altura de un pino y devuelve su peso.
- Definir la función **esPesoUtil**, recibe un peso en kg y devuelve True si un pino de ese peso le sirve a la fábrica, y False en caso contrario.
- Definir la función **sirvePino**, recibe la altura de un pino y devuelve True si un pino de ese peso le sirve a la fábrica, y False en caso contrario. Usar composición en la definición.

12. Este ejercicio alguna vez se planteó como un **Desafío Café con Leche**: Implementar la función **esCuadradoPerfecto/1**, sin hacer operaciones con punto flotante. Ayuda: les va a venir bien una función auxiliar, tal vez de dos parámetros. Pensar que el primer cuadrado perfecto es 0, para llegar al 2do (1) sumo 1, para llegar al 3ro (4) sumo 3, para llegar al siguiente (9) sumo 5, después sumo 7, 9, 11 etc.. También algo de recursividad van a tener que usar.