

Ejercicios Adicionales de Listas

1. Escribe la función **OrdenaRepetidos**, a la que le pasas una lista con números y te la ordena en función del número de veces que se repiten sus elementos, con los números que más se repiten apareciendo en las primeras posiciones y los que menos se repiten (o aparecen una sola vez) apareciendo en las últimas.

Ej.: [5, 5, 1, 2, 8, 3, 0, 3, 2, 2, 5, 7, 5] => [5, 5, 5, 5, 2, 2, 2, 3, 3, 1, 8, 0, 7]

2. Escribe las funciones **ElementosUnicos** y **ElementosRepetidos**. La función *ElementosUnicos* recibirá una lista y nos devolverá otra lista que contendrá solamente los elementos de la primera que sólo aparecen una vez.

Ej.: {1, 1, 2, 3, 99, 1, 99, 5} nos devolvería {2, 3, 5}

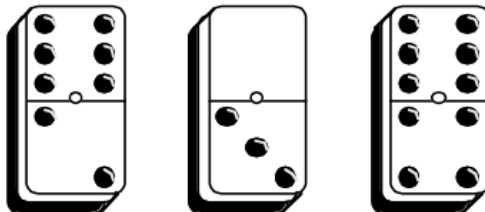
La función *ElementosRepetidos* hace justo lo contrario: recibirá una lista y nos devuelve otra lista con los elementos que aparecen repetidos en la primera lista.

Ej.: {1, 1, 2, 3, 99, 1, 99, 5} nos devolvería {1, 99}

Las funciones no deberán modificar la lista que se les pasa por parámetro.

3. Escribe la función **ListaDominoDobles** a la que le pasaremos una lista de enteros que representará fichas de dominó. La representación será la siguiente: cada pareja de enteros se corresponderá con una ficha, estando cada valor comprendido entre 0 y 6.

Ej.:



Estas tres fichas se representarían como la lista:

< 6, 2, 0, 3, 6, 4 >

La función nos devolverá una lista de enteros en la que aparecerán las fichas dobles (las que tienen los dos valores iguales) que aparezcan en la lista que le pasamos.

Ej.: Si le pasamos < 1, 1, 2, 6, 0, 0, 5, 4 > nos devolvería < 1, 1, 0, 0 >

La función deberá comprobar también si la lista con fichas de dominó que le pasamos es correcta y dará un mensaje de error en caso contrario. En concreto, habrá que comprobar que ninguno de los valores esté fuera del rango 0-6 y que el número de valores sea par (porque cada ficha está compuesta por dos valores).

4. Escribe la función **PalabrasMismaLetra** a la que le pasamos una cadena y nos devolverá una lista de cadenas que contendrá las palabras que empiezan y acaban por la misma letra. A la hora de contar las palabras, habrá que ignorar los símbolos de puntuación y las mayúsculas y minúsculas.

Ej.: "Amanda tiene tres serpientes" devolvería {amanda, serpientes}

5. Escribe la función **ListaPalabrasLargas**, a la que le pasamos una cadena que contiene palabras separadas por espacios y un entero en el que le indicamos la longitud mínima de una palabra y nos devuelve una lista de cadenas que contendrá solamente las palabras que tienen una longitud igual o mayor a la mínima.

Ej.: "Mi mamá me mima" (longitud = 4) devolvería {mamá, mima}

6. Escribe la función **HaceCalor** a la que la pasamos dos listas: una lista de *string* con nombres de localidades y una lista de *double* con temperaturas. Cada temperatura corresponde a la localidad que está en la misma posición de la otra lista.

Ej.: [Jerez, Tarifa, Barbate]

[18.9, 14.1, 18.6]

La función deberá calcular la temperatura media y borrar de las listas las localidades y temperaturas que estén por debajo de la media (es decir, dejar sólo las temperaturas que estén por encima de la media).

En el ejemplo, habría que borrar "Tarifa" y su temperatura "14.1".

Si las listas tienen diferente tamaño, deberá dar un error.