Functional Programming in Erlang

The aim of these exercises is to familiarise you with other ways of defining functions over lists in Erlang, in particular the different way that recursive functions can *construct* lists.

Transforming list elements

Define an Erlang function `double/1` to double the elements of a list of numbers.

Filtering lists

Define a function `evens/1` that extracts the even numbers from a list of integers.

Going further

If you want to try some other recursions on lists try to define functions to give:

- the `median` of a list of numbers: this is the middle element when the list is ordered (if the list is of even length you should average the middle two)
- the `modes` of a list of numbers: this is a list consisting of the numbers that occur most frequently in the list; if there is is just one, this will be a list with one element only

In doing this you might find it useful to think of other functions that you could define to help you solve these problems, such as a function to sort a list, or to work out how many times a value occurs in a particular list.

Once you have written your solutions, particularly for the latter questions, you might like to discuss your approach to solving the problems with other participants using the comments on this step.
