Jutge.org
The Virtual Learning Environment for Computer Programming

Haskell — Usage of higher-order functions (1) P93632_en

Implement the following functions using higher-order functions (and other predefined functions) of Haskell without using recursion.

1. Implement a function *eql* :: [**Int**] → [**Int**] → **Bool** that tells wether two lists of integers are equal.
2. Implement a function *prod* :: [**Int**] → **Int** that returns the product of a list of integers.
3. Implement a function *prodOfEvens* :: [**Int**] → **Int** that returns the product of all even numbers of a list of integers.
4. Implement a function *powersOf2* :: [**Int**] that generates the list of all the powers of 2.
5. Implement a function *scalarProduct* :: [**Float**] → [**Float**] → **Float** that returns the dot product of two lists of float numbers with the same size.

Scoring

Each function scores 20 points.

Sample input

```
eql [1,2,3] [1,2,3]
eql [1,2,3] [3,2,1]
eql [1,2,3] [1,2,3,4]
prod [2,10,5]
prodOfEvens [2,10,5]
take 5 powersOf2
scalarProduct [2.0,1.0,5.0] [3.0,2.0,2.0]
```

Sample output

```
True
False
False
100
20
[1,2,4,8,16]
18.0
```

Problem information

Author : Albert Rubio / Jordi Petit
Translator : Jordi Petit
Generation : 2016-01-19 11:28:40