

MC346 - Paradigmas de Programação

Prova Haskell - 24/10/2017

Em cada questão abaixo, você pode definir funções auxiliares à vontade, conforme achar necessário.

Questão 1 (Valor 2,5) Escreva um programa `conta.hs` em Haskell que continuamente lê uma linha da entrada padrão e imprime o número de caracteres e o número de palavras nesta linha.

Exemplo:

```
% runhaskell conta.hs
la la la
caracteres= 8 palavras= 3
dubi chu
caracteres= 8 palavras= 2

%
```

Questão 2 (Valor 2,5) Escreva uma função `mesmaSoma` em Haskell que recebe uma lista de listas de inteiros e retorna um valor booleano, que é verdadeiro quando todas as listas da lista de entrada possuem a mesma soma.

Exemplos:

```
> mesmaSoma [[2,3], [5]]
True
> mesmaSoma [[1,1], [2,2,2]]
False
> mesmaSoma [[-8], [-4,-4], [8,12,-28]]
True
```

Nota: a lista vazia satisfaz ao critério de mesma soma. Uma lista vazia tem soma zero.

Questão 3 (Valor 2,5) Considere a seguinte definição de árvore em Haskell:

```
data Tree = Nil | Node String (Tree) (Tree)
```

Escreva uma função `insere s t` que recebe uma cadeia de caracteres (`String`) `s` e uma árvore `t` como acima e retorna uma nova árvore com a cadeia inserida no lugar correto. Suponha que a árvore de entrada é uma árvore de busca. Se a cadeia dada já existir na árvore, a sua função deve retornar uma árvore igual à original.

Questão 4 (Valor 2,5) Escreva uma função `part k n` que recebe dois inteiros e retorna uma lista com todas as listas `p` de inteiros que satisfazem as condições:

- a soma dos elementos de `p` dá `n`;
- a lista `p` só possui elementos positivos (não tem zeros nem negativos);
- a lista `p` está em ordem não-decrescente;
- a lista `p` possui apenas elementos maiores ou iguais a `k`.

A lista retornada não pode ter listas repetidas, mas a ordem em que aparecem não importa.

Exemplos:

```
> part 3 5
[[5]]
> part 2 5
[[2,3],[5]]
> part 1 4
[[1,1,1,1],[1,1,2],[1,3],[2,2],[4]]
> part (-1) 3
[[1,1,1],[1,2],[3]]
```

Boa sorte!