

Universidade Federal de Uberlândia
Faculdade de Computação
Professor: Wendel Melo

Segunda lista de exercícios de Programação Funcional (2017-1)

Questão 1: Escreva uma função recursiva em Haskell que receba uma lista de números inteiros e retorne o menor número presente na lista. Atenção: você não pode utilizar a função *minimum*.

Questão 2: Escreva uma função recursiva em Haskell que receba uma lista de números *Float* e retorne o produto dos elementos da lista. Atenção: você não pode utilizar a função *product*.

Questão 3: Escreva uma função recursiva em Haskell que receba uma lista de *Int* e retorne *True* se **todos** os elementos da lista forem números pares, e *False*, caso contrário.

Questão 4: Escreva uma função recursiva em Haskell que receba um número inteiro *a* e uma lista *p* de números inteiros e retorne *True* se *a* estiver em *p*, e *False* caso contrário. Atenção: não é permitido o uso da função *element*.

Questão 5: Escreva uma função recursiva em Haskell que receba uma lista de caracteres *p* (*String*) e retorne uma nova lista onde cada caracter de *p* é substituído pelo caracter imediatamente subsequente na tabela ASCII. Assuma que *p* é composta apenas por caracteres alfabéticos e espaço.

Questão 6: Escreva uma função recursiva em Haskell que receba uma lista de inteiros *p* e retorna uma lista *q* com os elementos de *p* em ordem invertida.

Questão 7: Escreva uma função recursiva em Haskell que receba uma lista de inteiros e retorne *True* se houver algum número ímpar na lista, e *False* contrário.