

Trabalho 1 (2023/2)

Disciplina de Programação Funcional

A) Data de entrega: 29/09/2023

B) Objetivo:

O objetivo deste trabalho é consolidar o conhecimento sobre conceitos e construção de funções recursivas em Haskell.

C) Enunciado:

Estamos interessados na implementação de diversas funções de manipulação de “valores binários”. Em todos os exercícios que lidam com representação em binário, considere que um número binário é armazenado como uma lista de zeros e uns. Utilize as listas do Haskell nestas soluções: `[Int]`. Para cada exercício apresente também um exemplo de chamada da função implementada. Cuidado para a interpretação da codificação binária solicitada em cada exercício, sugere-se fortemente que sejam revisados os conceitos de representação numérica em binário antes de realizar a atividade!

1. Definir uma função recursiva que recebe um número binário (interpretado como número inteiro sem sinal) e retorna o valor equivalente em decimal. $bin2dec :: [Int] \rightarrow Int$
2. Definir uma função recursiva que recebe um número decimal inteiro não-negativo, um número de bits desejado e retorna o valor equivalente em binário (interpretado como número inteiro sem sinal) com o número de bits informado. Por exemplo, $dec2bin\ 2\ 8$ deve retornar `[0,0,0,0,0,0,1,0]`. $dec2bin :: Int \rightarrow Int \rightarrow [Int]$
3. Definir uma função recursiva que recebe um número binário na representação de complemento de dois e retorna o valor equivalente em decimal inteiro. $bincompl2dec :: [Int] \rightarrow Int$
4. Definir uma função recursiva que recebe um número decimal inteiro, um número de bits desejado e retorna o valor equivalente em binário na representação de complemento de dois com o número de bits informado. Por exemplo, $dec2bincompl\ (-2)\ 8$ deve retornar `[1,1,1,1,1,1,1,0]` $dec2bincompl :: Int \rightarrow Int \rightarrow [Int]$
5. Definir uma função recursiva que recebe um número fracionário decimal por parâmetro e devolve um número binário de ponto fixo de 32 bits. O número binário de ponto fixo deve ser representado por uma tupla com dois números binários tal que a parte inteira deve estar na representação de complemento de dois com 16 bits e a parte fracionária deve estar na representação de binário fracionado com 16 bits. Você deve definir uma forma adequada de representar o resultado caso o número decimal estoure a representação. Por exemplo, $frac2bin\ (-8.5)$ deve retornar `([1,1,1,1,1,1,1,1,1,1,1,1,0,0,0], [1,0,0,0,0,0,0,0,0,0,0,0,0,0,0])`. $frac2bin :: Double \rightarrow ([Int], [Int])$
6. Definir uma função recursiva que recebe uma tupla com dois números binários representando, respectivamente, a parte inteira (na representação de complemento de dois com 16 bits) e a parte fracionária (na representação de binário fracionado com 16 bits) de um número binário de ponto fixo com 32 bits, e retorna o correspondente valor fracionário decimal. Por exemplo, $bin2frac\ ([0,1,0,0,0,0,0,0,0,0,0,0,1,0,0,0], [1,0,1,0,0,0,0,0,0,0,0,0,0,0,0])$ deve retornar `16392.625`. $bin2frac :: ([Int], [Int]) \rightarrow Double$

D) Desenvolvimento e avaliação do trabalho:

- O trabalho pode ser realizado individualmente ou em grupos de, no máximo, 3 alunos.
- Programas que não consigam ser executados receberão nota zero.

- Mensagens de erro apresentadas durante a execução do programa serão consideradas como erros de execução, e acarretarão descontos na nota do trabalho.
- Os trabalhos serão avaliados de acordo com critérios a serem estabelecidos pelo professor da disciplina, considerando o que é pedido no enunciado e o que foi realizado com sucesso.
- ***Trabalhos copiados resultarão em nota zero para todos os alunos envolvidos.***

E) Entrega do trabalho:

- Todos os arquivos-fonte deverão ser empacotados em um único arquivo (.zip) e submetidos através do sistema Moodle até a data de entrega.
- Não serão aceitos trabalhos enviados por correio eletrônico.
- Não serão aceitos trabalhos enviados fora do prazo estabelecido.