

## LISTA DE EXERCICIOS 02 – UNIDADE 2

PROGRAMACAO IMPERATIVA E FUNCIONAL – TURMA A E B – 2021.1

- 
- **Deadline:** Segunda-feira, dia 01/11/2021, as 23:59.
  - **Conteúdo:** Intro a Haskell, Listas.
  - **Linguagem:** Haskell
  - Submeta todas as funções em um arquivo .hs único no formato L2\_login.hs (por exemplo, minha submissão seria L2\_ptlb.hs). Indique a qual questão cada função pertence através de comentários no código. **Qualquer arquivo enviado fora desse padrão será automaticamente descartado;**
  - Não use funções padrões do Haskell (como length, tail, sum, reverse, etc), exceto quando exigido claramente pela questão. **O uso indevido de funções padrões resultara na anulação da questão.**
  - O nome das funções implementadas **deve ser igual ao indicado pela questão**, caso contrário o script de correção não conseguira identificar sua questão e **ela será automaticamente zerada.**
  - Se o tipo da função não está claramente definido na descrição da questão, considere uma função que recebe uma lista de inteiros e retorna um inteiro ou lista de inteiros;
- 

- 1- Defina uma função *converteABC::Char -> Char* que converta letras minúsculas em letras maiúsculas. Esta função deve manter os caracteres que não são letras intactos (dica – use a Tabela ASCII);

EX: *converteABC* 'f' => 'F'

EX: *converteABC* '+' => '+'

- 2- Defina a função *romanDigit :: Char -> String* que converte um dígito de 1 a 10 em sua representação romana. A função mantém os números fora deste intervalo intactos.

EX: *romanDigit* '7' => "VII"

EX: *romanDigit* '0' => "0"

- 3- Defina uma função recursiva *digits :: String -> String* que filtra dígitos de uma String fornecida;

EX: *digits* "a1b23c" => "123"

EX: *digits* "DF34IO9" => "349"

- 4- Defina uma função recursiva ***newReverse::[Int] -> [Int]*** que retorna o inverso de uma lista xs;

EX: newReverse [1,2,3,4,5] => [5,4,3,2,1]

EX: newReverse [] => []

EX: newReverse [5] => [5]