

Paradigmas de Linguagens de Programação

Bacharelado em Ciência da Computação

Prof. Dr. Eduardo Takeo Ueda

2ª Lista de Exercícios

1. Escreva uma função (em Haskell) chamada `bissexto` que, dado um ano, verifica se o mesmo é bissexto.

Exemplo:

```
Main> bissexto 2002  
False
```

2. Construa uma função (em Haskell) chamada `mdc3n` que calcule o valor do máximo divisor comum entre três números naturais positivos.

Exemplo:

```
Main> mdc3n 2 3 4  
1
```

3. Escreva uma função (em Haskell) chamada `unicaOcorrencia` que recebe um elemento e uma lista e verifica se existe uma única ocorrência do elemento na lista.

Exemplo:

```
Main> unicaOcorrencia 2 [1,2,3,2]  
False
```

4. Escreva uma função (em Haskell) chamada `maioresQue` que recebe um número e uma lista de números, e retorna uma lista com os números que são maiores do que o valor informado.

Exemplo:

```
Main> maioresQue 10 [4 6 30 3 15 3 10 7]  
[30 15]
```

5. Escreva uma função (em Haskell) chamada `kElemento` que recebe um índice inteiro e uma lista, e retorna o k-ésimo elemento desta lista, caso ele exista.

Exemplo:

```
Main> kElemento 2 [1,3,7,9]
3
```

6. Escreva uma função (em Haskell) recursiva chamada `concatena` que recebe duas listas quaisquer e retorna uma terceira lista com os elementos da primeira no início e os elementos da segunda no fim.

Exemplo:

```
Main> concatena [1 2] [3 4]
[1 2 3 4]
```

7. Escreva uma definição recursiva da função (em Haskell) `prefixo :: String -> String -> Bool` que verifica se uma cadeia de caracteres é prefixo de outra.

Exemplo:

```
Main> prefixo "abra" "abracadabra"
True
```

8. Escreva um programa (em Haskell) que ordena uma lista aplicando o algoritmo de ordenação por seleção.

Exemplo:

```
Main> selectionSort [2,3,4,5,2,3,2,4,5,3,2,2]
[2,2,2,2,2,3,3,3,4,4,5,5]
```

9. Escreva um programa (em Haskell) que ordena uma lista aplicando o algoritmo de ordenação por inserção.

Exemplo:

```
Main> insertionSort "brasil"
"abilrs"
Main> insertionSort [4,3,67,23,5]
[3,4,5,23,67]
```

10. Escreva um programa (em Haskell) que ordena uma lista aplicando o algoritmo Merge Sort.

Exemplo:

```
Main> mergeSort [3,4,1,2]
```

```
[1,2,3,4]
```

```
Main> mergeSort "LPG3"
```

```
"3GLP"
```