

CURSO: Ciência da Computação	CÓDIGO: CI062
PERÍODO: 4o.	PROFESSOR: Andrey
DISCIPLINA: Técnicas Alternativas de Programação	AULA: 18 Exercícios

1 Apresentação

Na aula de hoje vamos exercitar os conceitos de manipulação de listas em Haskell. A aula de hoje é baseada em [de Sá et al., 2006]

2 Atividades

- 1: Construa uma função que inverta uma lista.
- 2: Construa uma função que remova o primeiro elemento de uma lista.
- 3: Construa uma função que remova o último elemento de uma lista.
- 4: Construa uma função que some os elementos pares de uma lista.
- 5: Construa uma função que some os elementos em posições pares de uma lista. (A primeira posição da lista é a posição 1).
- 6: Construa uma função que receba uma lista e um número 'n' e insira um elemento na n-ésima posição da lista.
- 7: Construa uma função que encontre o penúltimo elemento de uma lista.
- 8: Construa uma função que elimine duplicatas consecutivas dos elementos da lista.
- 9: Construa uma função que divida uma lista em duas partes; é fornecido o comprimento da primeira parte.
- 10: Construa uma função que dados dois índices, i e k, a fatia é a lista que contém os elementos entre os elementos i-ésimo e o k-ésimo da lista original (ambos os limites incluídos). Comece a contar os elementos com 1.
- 11: Construa uma função que gire uma lista N lugares para a esquerda.
- 12: Construa uma função que crie uma lista contendo todos os números inteiros dentro de um determinado intervalo.

In English:

- 1: Create a function that inverts a list.

- 2:** Create a function that removes the first element from a list.
- 3:** Create a function that removes the last element from a list.
- 4:** Create a function that adds the even elements of a list.
- 5:** Create a function that sums the elements at even positions of a list. (The first position in the list is position 1).
- 6:** Create a function that takes a list and a number 'n' and inserts an element in the nth position of the list.
- 7:** Create a function that finds the second to last element in a list.
- 8:** Create a function that eliminates consecutive duplicates of list elements.
- 9:** Create a function that divides a list into two parts; The length of the first part is provided.
- 10:** Create a function that, given two indices, i and k, the slice is the list that contains the elements between the ith and kth elements of the original list (both bounds included). Start counting the elements with 1.
- 11:** Create a function that rotates N places to the left.
- 12:** Create a function that creates a list containing all integers within a given range.

Referências

- [de Sá et al., 2006] de Sá, C. C., da Silva, and Ferreira, M. (2006). *Haskell: uma abordagem prática*. Novatec Editora.