

1. Declare uma função que verifica se um elemento pertence a uma lista; a função deve retornar True se o elemento estiver na lista, e False caso contrário.

Exemplo: pertence 3 [1, 4, 3, 2] => True

2. Declare uma função que retorne a interseção entre duas listas.

Exemplo: intersecao [1, 3, 5, 7, 9] [2, 5, 3, 6, 9] => [3, 5, 9]

3. Declare uma função que retorne o inverso de uma lista.

Exemplo: inverso [1, 2, 3, 4] => [4, 3, 2, 1]

4. Declare uma função que retorne os n últimos elementos de uma lista.

Exemplo: nUltimos 3 [1, 2, 3, 4, 5, 6] => [4, 5, 6]

5. Declare uma função que receba duas listas de números e crie uma lista com a soma do primeiro elemento da primeira lista com o primeiro elemento da segunda lista, a soma do segundo elemento da primeira lista com o segundo elemento da segunda lista, e assim sucessivamente até que uma das listas termine.

Exemplo: soma2 [1, 2, 3, 4] [10, 20, 30] => [11, 22, 33].

6. Declare uma função que receba como parâmetro um número n e retorne uma lista com todas as potências de 2 até 2^n .

Exemplo: pot2 4 => [2, 4, 8, 16]

7. Declare uma função que receba duas listas previamente ordenadas e faça a intercalação (*merge*) dos elementos tendo como resultado a junção das duas listas em uma lista também ordenada, *sem reordenar nenhuma das listas*.

Exemplo: intercalacao [10, 15, 17, 20] [1, 2, 13, 15, 22] => [1, 2, 10, 13, 15, 15, 17, 20, 22]

8. Declare uma função que retorne o menor elemento de uma lista.

Exemplo: menor [10, 3, 5, 2, 20] => 2

9. Declare uma função que receba uma lista e um elemento e retorne a lista sem a primeira ocorrência desse elemento.

Exemplo: removerElem 1 [2, 4, 1, 3, 2, 1] => [2, 4, 3, 2, 1]

10. Usando as declarações anteriores (menor e removerElem), declare uma função que ordene os elementos de uma lista.

Exemplo: ordenar [32, 10, 23, 10, 12, 4] => [4, 10, 10, 12, 23, 32]

11. Declare uma função que receba um elemento e uma lista ordenada, e insira este elemento na lista o colocando na posição correta, ou seja, a lista resultante deve estar ordenada. Se o elemento já pertencer à lista, ele não deve ser incluído, e a lista não deve ser reordenada.

Exemplo: `insereOrd 12 [6, 9, 10, 15, 20] => [6, 9, 10, 12, 15, 20]`

12. Declare uma função que receba um número n e uma lista, e retorne o n -ésimo elemento.

Exemplo: `enesimo 3 [10, 20, 30, 40, 50] => 30`

13. Declare uma função que receba um inteiro n e um elemento e e crie uma lista com n elementos e .

Exemplo: `repetir 4 10 => [10, 10, 10, 10]`

14. Declare uma função que troque todos os caracteres de tabulação (`'\t'`) por espaços em uma String.

Exemplo: `removeTab "1\tTeste" => "1 Teste"`

15. Declare uma função que receba uma String e converta todas letras maiúsculas dessa String em letras minúsculas.

Exemplo: `minusculas "AbCdeF" => "abcdef"`

16. Declare uma função que receba uma lista de duplas e retorne uma lista com o inverso de cada dupla.

Exemplo: `inversoDupla [(1, 2), (6, 1), (4, 11)] => [(2, 1), (1, 6), (11, 4)]`

17. Declare uma função que receba uma lista de duplas, e retorne lista de booleanos indicando se os elementos são iguais ou não.

Exemplo: `simetrico [(1, 2), (4, 4), (3, 2)] => [False, True, False]`

18. Declare uma função que converta um inteiro em um número inteiro em formato de String.

Exemplo: `numString 126 => "126"`

19. Declare uma função que converta uma String contendo uma sequência de dígitos para um inteiro, ou seja, o inverso da questão anterior.

Exemplo: `stringNum "102" => 102`

20. Declare uma função que converta um inteiro em um número binário, o representado como uma String.

Exemplo: `decBin 11 => "1011"`

Declare uma função que converta um número binário (representado como uma String) em um número inteiro.

Exemplo: binDec "1011" => 11

22. Desenvolva uma função em Haskell que permita calcular o troco em moedas para o café. Para isso, a função deve receber o valor do café (Int) e o valor em dinheiro pago pelo cliente (Int), e retornará uma lista de tuplas [(a, b)], tal que a é o valor da moeda, e b a quantidade de moedas deste valor. São permitidas moedas de 5, 10, 20 e 50 centavos, e deve ser sempre retornado moedas de maior valor antes.

Exemplo: trocoCafe 65 110 => [(20,2), (5,1)]