

Trabalho Extra – LINGUAGEM GHERKIN

EXERCÍCIO 4

Funcionalidade: Testar o Método somarElementos da Classe ListaNumerica

Cenário: Somando elementos de uma lista

Dado: Uma lista de números [1, 4, 5]

Quando: Eu somar os elementos da lista com o método somarElementos da classe ListaNumerica

Então: o resultado deve ser 10

Cenário: Somando elementos de uma lista com 0

Dado: Uma lista de números [1, 4, 0, 5]

Quando: Eu somar os elementos da lista com o método somarElementos da classe ListaNumerica

Então: o resultado deve ser 10

E: o zero deve ser ignorado

Cenário: Somando elementos de uma lista negativa

Dado: Uma lista de números [-1, -4, -5]

Quando: Eu somar os elementos da lista com o método somarElementos da classe ListaNumerica

Então: o resultado deve ser -10

Cenário: Somando elementos de uma lista vazia

Dado: Uma lista de números vazia

Quando: Eu somar os elementos da lista com o método somarElementos da classe ListaNumerica

Então: o resultado deve ser 0

Cenário: Somando elementos de uma lista com números float

Dado: Uma lista de números [1.1, -1.1]

Quando: Eu somar os elementos da lista com o método
somarElementos da classe ListaNumerica

Então: o resultado deve ser 0

Funcionalidade: Testar o Método encontrarMaiorElemento da
Classe ListaNumerica

Cenário: Encontrando o maior elemento de uma lista

Dado: Uma lista de números [3,4,5]

Quando: Eu encontrar o maior elemento da lista com o método
encontrarMaiorElemento da classe ListaNumerica

Então: o resultado deve ser 5

E: não deve retornar 3 ou 4

Cenário: Encontrando o maior elemento de uma lista com
negativos e positivos

Dado: Uma lista de números [-1,0,2]

Quando: Eu encontrar o maior elemento da lista com o método
encontrarMaiorElemento da classe ListaNumerica

Então: o resultado deve ser maior que 1

E: não deve retornar 0 ou -1

Mas: deve retornar o 2

Cenário: Encontrando o maior elemento de uma lista vazia

Dado: Uma lista de números vazia

Quando: Eu encontrar o maior elemento da lista com o método
encontrarMaiorElemento da classe ListaNumerica

Então: o resultado deve ser falso

Cenário: Encontrando o maior elemento de uma lista com números float

Dado: Uma lista de números [1.1, -1.1]

Quando: Eu encontrar o maior elemento da lista com o método encontrarMaiorElemento da classe ListaNumerica

Então: o resultado deve ser 1.1

E: não deve retornar -1.1

Funcionalidade: Testar o Método encontrarMenorElemento da Classe ListaNumerica

Cenário: Encontrando o menor elemento de uma lista

Dado: Uma lista de números [3,4,5]

Quando: Eu encontrar o menor elemento da lista com o método encontrarMenorElemento da classe ListaNumerica

Então: o resultado deve ser 3

E: não deve retornar 4 ou 5

Cenário: Encontrando o menor elemento de uma lista com negativos e positivos

Dado: Uma lista de números [-1,0,2]

Quando: Eu encontrar o menor elemento da lista com o método encontrarMenorElemento da classe ListaNumerica

Então: o resultado deve ser menor que 0

E: não deve retornar 1 ou 2

Mas: deve retornar o -1

Cenário: Encontrando o menor elemento de uma lista vazia

Dado: Uma lista de números vazia

Quando: Eu encontrar o menor elemento da lista com o método encontrarMenorElemento da classe ListaNumerica

Então: o resultado deve ser falso

Cenário: Encontrando o menor elemento de uma lista com números float

Dado: Uma lista de números [1.1, -1.1]

Quando: Eu encontrar o menor elemento da lista com o método encontrarMenorElemento da classe ListaNumerica

Então: o resultado deve ser -1.1

E: não deve retornar 1.1

Funcionalidade: Testar o Método ordenarLista da Classe ListaNumerica

Cenário: Ordenando uma lista

Dado: Uma lista de números [4, -1, 0]

Quando: Eu ordenar a lista com o método ordenarLista da classe ListaNumerica

Então: o resultado deve ser [-1, 0, 4]

E: o resultado deve ser uma array

E: o resultado deve ter o mesmo tamanho da lista original

Mas: não deve retornar a lista original

Cenário: Ordenando uma lista vazia

Dado: Uma lista de números vazia

Quando: Eu ordenar a lista com o método ordenarLista da classe ListaNumerica

Então: o resultado deve ser uma array vazia

E: o resultado deve ter o mesmo tamanho da lista original

Funcionalidade: Testar o Método filtrarNumerosPares da Classe ListaNumerica

Cenário: Filtrando números pares de uma lista

Dado: Uma lista de números [1, -2, 0]

Quando: Eu filtrar os números pares da lista com o método
filtrarNumerosPares da classe ListaNumerica

Então: o resultado deve ser [-2, 0]

E: o resultado deve ser uma array

E: o resultado deve ter o mesmo tamanho da lista original

E: o resultado deve ter apenas números pares

Mas: o resultado deve ser diferente da lista original

Cenário: Filtrando números pares de uma lista vazia

Dado: Uma lista de números vazia

Quando: Eu filtrar os números pares da lista com o método
filtrarNumerosPares da classe ListaNumerica

Então: o resultado deve ser uma array vazia

Cenário: Filtrando números pares de uma lista com apenas
números ímpares

Dado: Uma lista de números [-1,1]

Quando: Eu filtrar os números pares da lista com o método
filtrarNumerosPares da classe ListaNumerica

Então: o resultado deve ser uma array vazia

Cenário: Filtrando números pares de uma lista com números float

Dado: Uma lista de números [1.1, -1.1]

Quando: Eu filtrar os números pares da lista com o método
filtrarNumerosPares da classe ListaNumerica

Então: o resultado deve ser uma array vazia