

Lista de Exercícios Extra – Listas

Exercício 1

Escreva uma função chamada `nested_sum` que receba uma lista de listas de números inteiros e adicione os elementos de todas as listas aninhadas. Por exemplo:

```
>>> t = [[1, 2], [3], [4, 5, 6]]
>>> nested_sum(t)
21
```

Exercício 2

Escreva uma função chamada `cumsum` que receba uma lista de números e retorne a soma cumulativa; ou seja, uma nova lista onde o elemento com índice *i* é a soma dos primeiros *i+1* elementos da lista original. Por exemplo:

```
>>> t = [1, 2, 3]
>>> cumsum(t)
[1, 3, 6]
```

Exercício 3

Escreva uma função chamada `middle` que receba uma lista e retorne uma nova lista com todos os elementos originais, exceto o primeiro e o último elementos. Por exemplo:

```
>>> t = [1, 2, 5, 7, 3, 4]
>>> middle(t)
[2, 5, 7, 3]
```

Exercício 4

Escreva uma função chamada `chop` que tome uma lista alterando-a para remover o primeiro e o último elementos, e retorne `None`. Por exemplo:

```
>>> t = [1, 2, 3, 4]
```

```
>>> chop(t)
>>> t
[2, 3]
```

Exercício 5

Escreva uma função chamada `is_sorted` que tome uma lista como parâmetro e retorne `True` se a lista estiver classificada em ordem ascendente, e `False` se não for o caso. Por exemplo:

```
>>> is_sorted([1, 2, 2])
True
>>> is_sorted(['b', 'a'])
False
```