

Lista 01 - Prog 01 2024

Emilio Vital Brazil

Entrega: 29 de abril 2024

Todas as questões abaixo serão consideradas usando a sintaxes **Python 3.11**.

1. Nos seguintes codigos escreva a saída padrão esperada:

(a)

```
1  x = 1
2  myList = [10, 11, x, 13, "Ola", "casa", "carro", "pé"]
3  for z in myList:
4      print(z)
```

(b)

```
1  x = 2
2  myList = [10, 11, x, 13, "Ola", "casa", "carro", "pé"]
3  for i in range(len(myList)):
4      print(i)
```

(c)

```
1  x = 'OLA'
2  myList = [10, 11, x, 13, "Ola", "casa", "carro", "pé"]
3  i = 1
4  while myList[i] != "carro":
5      print(myList[i])
6      i+=1
```

2. Escreva uma função chamada *sum_cumulative* que receba uma lista de números inteiros e retorne uma nova lista contendo a soma cumulativa dos elementos da lista de entrada. A soma cumulativa de um elemento é a soma dele com todos os elementos anteriores. Por exemplo:

```
>>> sum_cumulative([-2, 3, 4, -1, 1])
[-2, 1, 5, 4, 5]
```

3. Escreva uma função chamada *nested_sum* que recebe 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
```

4. Escreva uma função chamada *relatively_prime* que recebe dois números inteiros e retorne o valor **True** caso os números sejam primos entre si e **False** caso contrário.
5. Escreva uma função chamada *before_half* que recebe um número inteiro como entrada e imprime todos os números inteiros entre o número fornecido e sua metade, exclusivamente.

Por exemplo:

```
>>> before_half(10)
10
9
8
7
6
```