

Lista 7**Listas (parte 1)****Questão 1**

As listas servem para guardar uma **coleção de objetos**, que podem ser números, palavras, etc. Quais outros nomes são usados para se referir a listas?

- (a) grupo
- (b) vetor
- (c) coleção
- (d) array
- (e) conjunto

Questão 2

Assinale as linhas que criam corretamente uma lista:

- (a) `idades = ["Frankfurt", "New York", "Tokio"]`
- (b) `idades = {4, 66, 34, 12, 1, 88}`
- (c) `flores = [margarida, rosa, tulipa, cravo]`
- (d) `aluno = ["Fulano de Tal", 25, "Rua xyz, 123", "São Paulo", "Matemática", 7.5, "Português", 6.6]`
- (e) `notas = [3.4, 10, 9.7, 5, 5.9]`
- (f) `carros = {"Ferrari", "Fusca", "Camry"}`
- (g) `horas = 1, 2, 6, 9, 10`

Questão 3

Escreva um programa em **Python** dada uma lista com valores de temperatura dos últimos 10 dias `temp = [20, 30, 28, 17, 23, 10, 11, 8, 39, 30]`, informe:

- (a) A temperatura do primeiro dia?
- (b) A temperatura do último dia?
- (c) O maior valor em `temp`.
- (d) A posição do maior valor em `temp`.
- (e) O menor valor em `temp`.

- (f) A posição do menor valor em `temp`.
- (g) O valor médio em `temp`

Questão 4

Qual(is) das seguintes afirmações estão **incorretas**?

- (a) Em **Python**, uma lista pode conter elementos de **diferentes tipos**.
- (b) A função `len()` retorna a quantidade de elementos em uma lista.
- (c) Podemos adicionar um elemento no **início** de uma lista com a função `append()`.
- (d) Para inicializar uma lista `l` vazia, utilizamos o comando `l = []`
- (e) Podemos alterar qualquer valor de uma lista.

```
1 l = [1, 2, 3]
2 l[1] = 10
```

- (f) O **Python** retorna `None` quando tentamos acessar uma posição inexistente na lista. Exemplo:

```
1 l = [1, 2, 3]
2 l[5] = 10
```

- (g) Objetos em uma lista **precisam** ser **únicos**.

```
1 l = ['a', 'b', 'c']
```

- (h) Em uma lista vazia `l`, podemos salvar o valor `'a'` no primeiro elemento com o comando `l[1] = 'a'`. Exemplo:

```
1 l = []
2 l[0] = 'a'
```

Questão 5

Qual será o resultado apresentado pelo programa abaixo?

```
1 flores = ["margarida", "rosa", "tulipa", "cravo"]
2 print(flores)
3
4 tam = len(flores) - 1
5 while tam >= 0:
6     print(flores[tam])
7     tam = tam - 1
```

Questão 6

Escreva um programa em **Python** que leia do teclado uma sequência de valores inteiros $a_1, a_2, a_3 \dots$ até que o usuário digite `0`. Em seguida imprima essa sequência na ordem inversa.

Questão 7

Escreva um programa em **Python** que leia dois vetores de **dimensão 3**, a e b e calcule o produto escalar destes.

$$a = (a_1, a_2, a_3) \text{ e } b = (b_1, b_2, b_3)$$

$$a \cdot b = \sum_{i=1}^3 a_i \times b_i$$

Questão 8

Escreva um programa em **Python** que leia duas listas com 5 inteiros cada e verifique **quais elementos** da primeira lista **são iguais** a algum elemento da segunda lista.

```
1 l1 = [1, 3, 5, 8, 9]
2 l2 = [5, 9, 10, 1, 0]
```

Resultado:

```
1 [5, 9, 1]
```

Se não houver elementos em comum, o programa deve informar isso.

Questão 9

Escreva um programa em **Python** que leia n números inteiros e os salva em uma lista. Em seguida o programa deve ler um outro **número inteiro C**. O programa deve então encontrar dois números de posições distintas da lista cuja multiplicação seja C e imprimi-los. Caso não existam tais números, o programa deve informar isto.

```
1 l = [2, 4, 5, -10, 7]
2 C = 35
```

Resultado:

```
1 5 e 7
```

Questão 10

Faça um programa em **Python** que leia duas sequências de n e m valores inteiros ordenados e salve cada sequência em uma lista. O seu programa deve intercalar os valores das duas listas em uma terceira lista, em ordem crescente.

```
1 l1 = [1, 3, 5, 5, 7, 9, 11, 12, 13]
2 l2 = [2, 2, 4, 6, 8, 8, 10]
```

Resultado:

```
1 l3 = [1, 2, 2, 3, 4, 5, 5, 6, 7, 8, 8, 9, 10, 10]
```