

1. Faça um programa que pede ao usuário 10 números e armazene-os em uma lista. Depois, peça outro número para o usuário e mostre quantos dos 10 números da lista são inferiores ao valor digitado.

```
--> 2
--> 4
--> 6
--> 8
--> 10
--> 12
--> 14
--> 16
--> 18
--> 20

Numero: 15
Dentre os numeros digitados, 7 sao menores que 15!
```

2. Faça um programa para:
  - a. receber 10 números inteiros do usuário e armazená-los em uma lista;
  - b. copiar os pares para uma lista de pares e os ímpares para o uma lista de ímpares, na ordem em que aparecem;
  - c. mostrar as listas resultantes no final.

```
--> 10
--> 9
--> 8
--> 7
--> 6
--> 5
--> 4
--> 3
--> 2
--> 1

Numeros: [10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
Pares: [10, 8, 6, 4, 2]
Impares: [9, 7, 5, 3, 1]
```

3. Faça um programa que peça 10 números ao usuário e adicione esses números a uma lista.

**Sem utilizar funções:**

- a. Mostre o **menor** número da lista;
- b. Mostre o **maior** número da lista.

**Utilize FOR ITEM IN LISTA e condicional para percorrer a lista e achar os números**

```
Digite o valor 1: 10
Digite o valor 2: 20
Digite o valor 3: -50
Digite o valor 4: 150
Digite o valor 5: 30
Digite o valor 6: 40
Digite o valor 7: 50
Digite o valor 8: 60
Digite o valor 9: 70
Digite o valor 10: 80

Números: [10, 20, -50, 150, 30, 40, 50, 60, 70, 80]
O maior número é: 150
O menor número é: -50
```

4. Crie uma lista com 10 valores aleatórios, mostre essa lista e depois organize ela de forma crescente **sem utilizar funções**.

**Utilize FOR e condicional**

```
Lista antes de organizar:
[14, 52, 92, 55, 59, 62, 4, 22, 32, 46]

Lista organizada em ordem crescente:
[4, 14, 22, 32, 46, 52, 55, 59, 62, 92]
```

5. Altere o exercício anterior para organizar a lista de forma decrescente.

**Utilize FOR e condicional**

```
Lista antes de organizar:  
[53, 42, 81, 53, 55, 6, 34, 89, 59, 66]
```

```
Lista organizada em ordem decrescente:  
[89, 81, 66, 59, 55, 53, 53, 42, 34, 6]
```

6. Crie duas listas com 10 valores aleatórios, e então compare as listas. Caso algum valor se repita em ambas, mostre na tela.

**Utilize FOR ITEM IN LISTA e condicional**

```
Lista 1:  
[1, 2, 3, 4, 5, 5, 6, 8, 1, 10]
```

```
Lista 2:  
[1, 2, 5, 10, 20, 25, 27, 0, 1, 2]
```

```
O número 1 está nas duas listas.  
O número 1 está nas duas listas.  
O número 2 está nas duas listas.  
O número 2 está nas duas listas.  
O número 5 está nas duas listas.  
O número 5 está nas duas listas.  
O número 1 está nas duas listas.  
O número 1 está nas duas listas.  
O número 10 está nas duas listas.
```

7. Incremente o código anterior para que caso um mesmo número se repita mais de uma vez, ele mostre uma única vez.

```
Lista 1:  
[1, 2, 3, 4, 5, 5, 6, 8, 1, 10]
```

```
Lista 2:  
[1, 2, 5, 10, 20, 25, 27, 0, 1, 2]
```

```
O número 1 está nas duas listas.  
O número 2 está nas duas listas.  
O número 5 está nas duas listas.  
O número 10 está nas duas listas.
```

8. Peça que o usuário digite 10 elementos (número, string etc.) e insira eles em uma lista. Depois, compare os elementos e verifique se algum deles se repete; se repetir, mostre ao usuário o elemento e suas posições na lista.

**Utilize FOR e condicional**

```
Digite o 1 elemento: a
Digite o 2 elemento: b
Digite o 3 elemento: 2
Digite o 4 elemento: f
Digite o 5 elemento: a
Digite o 6 elemento: b
Digite o 7 elemento: 0
Digite o 8 elemento: a
Digite o 9 elemento: b
Digite o 10 elemento: 2

Lista:
['a', 'b', '2', 'f', 'a', 'b', '0', 'a', 'b', '2']

Repetições:

0 elemento *a* se repete nas posições 1. 5. 8.
0 elemento *b* se repete nas posições 2. 6. 9.
0 elemento *2* se repete nas posições 3. 10.
```

-----

```
Digite o 1 elemento: a
Digite o 2 elemento: b
Digite o 3 elemento: c
Digite o 4 elemento: d
Digite o 5 elemento: e
Digite o 6 elemento: f
Digite o 7 elemento: g
Digite o 8 elemento: h
Digite o 9 elemento: i
Digite o 10 elemento: j

Lista:
['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j']

Repetições:
Nenhum elemento se repetiu na lista :/
```