



## MÓDULO 2. UNIDADE 5

Exercícios de Controlo de  
fluxo



### DIRETRIZES GERAIS

- Guardar o documento de soluções com o seguinte formato para entrega:  
**M2\_01\_nome\_apelido1\_apelido2**
- Software recomendado: **Anaconda** – Jupyter. Junto a este documento está um padrão de Jupyter com os enunciados
- **Comentar o código**
- Utilizar nomes de variáveis apropriados. Se vamos guardar uma nota, chamamos a essa variável nota, não n ou x.



## EXERCÍCIOS DE CONDICIONAIS E CICLOS

1. Fazer um programa que peça três números ao utilizador e faça as seguintes instruções:

- Informa se os números estão em ordem ascendente ou
- Informa se os números não estão em ordem ascendente e
- Deverá indicar que o primeiro número introduzido não pode ser 0 se for o caso.



### EXERCÍCIOS DE CONDICIONAIS E CICLOS

2. Fazer um programa que calcule o somatório dos primeiros N números, começando em 0 até N-1 e sendo N o número que o utilizador proporciona:

- Exemplo de número limite introduzido pelo utilizador: 5
- Somatório =  $0 + 1 + 2 + 3 + 4$  (somamos os 5 primeiros números)



### EXERCÍCIOS DE CONDICIONAIS E CICLOS

3. Fazer um programa que peça três números ao utilizador e crie o seguinte menu:

- Mostrar uma soma dos dois números.
- Mostrar uma subtração dos dois números (o primeiro menos o segundo).
- Mostrar uma multiplicação dos dois números.
- Em caso de não introduzir uma opção válida, o programa vai informar que não é correta.



### EXERCÍCIOS DE CONDICIONAIS E CICLOS

4. Fazer um programa que leia letras e conte para uma variável com o nome "counter" a quantidade de letras "a" que são introduzidas.

Para sair do programa, deverá ser introduzido o carácter "."

Ao finalizar deve mostrar o número de vezes que a letra "a" foi pressionada.



### EXERCÍCIOS DE CONDICIONAIS E CICLOS

5. Fazer um programa que leia por teclado um número impar.

Deve repetir-se o processo enquanto o utilizador continua a introduzir

números ímpares. Quando introduzir um número par, o programa

deve parar.



### EXERCÍCIOS DE CONDICIONAIS E CICLOS

6. Fazer um programa que some todos os números pares desde o 0 até ao 100:

*Sugestão: podemos utilizar as funções `sum()` e `range()` para o fazer mais fácil. O terceiro parâmetro na função `range` (início, fim, passo) poderá ajudar, experimente.*





### EXERCÍCIOS DE CONDICIONAIS E CICLOS

7. Fazer um programa que solicite ao utilizador um número que representa o total de números que se vai, de seguida, pedir ao utilizador.

Posteriormente, ler todos esses números e fazer a média aritmética de todos.



### EXERCÍCIOS DE CONDICIONAIS E CICLOS

8. Fazer um programa que solicite ao utilizador um número inteiro de 0 a 9, e que enquanto o número não seja nesse intervalo, deve-se repetir o processo. De seguida, deve verificar se o número introduzido se encontra na lista de "numeros" e notificá-lo positivamente ou negativamente caso não esteja:

*Dica: a sintaxe "valor in lista" permite verificar facilmente se um valor se encontra numa lista (devolver True ou False)*

```
números = [1, 3, 6, 9]
```



### EXERCÍCIOS DE CONDICIONAIS E CICLOS

9. Fazer um programa que utilize a função `range()` para gerar números, faça a conversão do output do `range()` para listas, e gere as seguintes listas dinamicamente (e mostrá-las com `print`):

- Todos os números do 0 ao 10 [0, 1, 2, ..., 10]
- Todos os números do -10 ao 0 [-10, -9, -8, ..., 0]
- Todos os números pares do 0 ao 20 [0, 2, 4, ..., 20]
- Todos os números ímpares entre -20 e 0 [-19, -17, -15, ..., -1]
- Todos os números múltiplos de 5 do 0 ao 50 [0, 5, 10, ..., 50]

*Dica: utilizar o terceiro parâmetro da função `range` (início, fim, passo).*



### EXERCÍCIOS DE CONDICIONAIS E CICLOS

10. Fazer um programa que dadas duas listas, construa uma terceira lista com todos os elementos que existam em ambas as listas, mas não se pode repetir nenhum elemento:

```
lista_1 = ['O','l','a',' ','m','u','n','d','o']
```

```
lista_2 = ['O','l','a',' ','l','u','a']
```

```
lista_3 = []
```