



MÓDULO 2. UNIDADE 5

Exercícios de Controlo de fluxo



DIRETRIZES GERAIS

- Guardar o documento de soluções com o seguinte formato para entrega:
M2_04_nome_apelido1_apelido2.
- Utilizar o Software: **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) Faça um programa que solicite ao utilizador três números individualmente e realize as seguintes verificações:
 - Se os números estão por ordem ascendente ou decrescente.
 - Se há um erro devido a que o primeiro número introduzido é 0.



EXERCÍCIOS DE CONDICIONAIS E CICLOS

2) Faça um programa que realize o somatório dos primeiros N números e apresente o valor da soma, onde N é fornecido pelo utilizador

- 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) Faça um programa que solicite ao utilizador 2 números e apresente as seguintes opções:

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



EXERCÍCIOS DE CONDICIONAIS E CICLOS

- 4) Faça um programa que solicite ao utilizador que introduza um número e o apresente.
- Repita o processo enquanto o utilizador continuar a introduzir números ímpares.
 - Quando o utilizador introduzir um número par, pare o programa.



EXERCÍCIOS DE CONDICIONAIS E CICLOS

5) Faça um programa que apresente a soma de todos os números pares desde 0 até 100:

- Pode utilizar as funções `sum()` e `range()` para o tornar mais fácil.
- O terceiro parâmetro na função `range` (início, fim, salto) indica um salto de números.



EXERCÍCIOS DE CONDICIONAIS E CICLOS

6) Crie um programa que solicite ao utilizador um número inteiro de 0 a 9. Repita o processo até que o número seja correto. Em seguida, verifique se o número está na lista específica (2, 4, 5, 7). Se estiver, informe ao utilizador; caso contrário, repita o processo.

Pista: a sintaxe “valor in lista” permite comprovar facilmente se um valor se encontra numa lista (devolver True ou False).



EXERCÍCIOS DE CONDICIONAIS E CICLOS

7) Faça um programa que utilize a função `range()` e a conversão para listas gere as seguintes listas dinamicamente (e mostrá-las com `print`):

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

Pista: utilizar o terceiro parâmetro da função `range` (início, fim, salto).



EXERCÍCIOS DE CONDICIONAIS E CICLOS

8) Crie um programa que, dadas duas listas, lista_1 e lista_2, gere uma terceira lista com os elementos que se repetem nas duas listas:

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

```
lista_2 = ["o", 'l', 'a', ' ', 'l', 'u', 'a']
```

Atenção: não deve repetir-se nenhum elemento na nova lista.