

## Fundamentos da Programação

## Aula 20

## PROJETO 1

Discussão 1o projeto. Boas práticas. Dúvidas. Testes unitários

ALBERTO ABAD, IST, 2022-23

Projeto

## Enunciado e testes

- Página da UC:

<https://fenix.tecnico.ulisboa.pt/disciplinas/FProg-2/2022-2023/1-semester/projetos>

- Enunciado:

<https://fenix.tecnico.ulisboa.pt/downloadFile/845043405604545/>

- Testes públicos:

<https://fenix.tecnico.ulisboa.pt/downloadFile/845043405604546/>

## Projeto

### Tres tarefas independentes

1. Justificação de texto
2. Método de Hondt
3. Solução de Sistemas de Equações

## Projeto

### Mooshack e avaliação automática (16 valores)

- Registro no Mooshak
- Submissão
- Testes automaticos e avaliação (*diff*)
- Problemas tipicos: encoding, deixar código, etc.
- Limite de submissões e *debugging*

## Avaliação manual (4 valores)

- Boas práticas (1.5 valores): clareza do código, integração de conhecimento adquirido durante a UC e a criatividade das soluções propostas.
- Comentários/documentação (1 valor): comentários para o utilizador (docstring) e assinatura de funções + comentários para o programador.
- Tamanho de funções, duplicação de código e abstração procedimental (1valor)
- Escolha de nomes (0.5 valores).
- Links úteis:
  - Recomendações:  
<https://gist.github.com/ruimaranhao/4e18cbe3dad6f6>
  - PEP-8 <https://www.python.org/dev/peps/pep-0008/>
  - PEP-257 <https://www.python.org/dev/peps/pep-0257/>

Avaliação manual: Exemplo *docstring*

```
In [3]: """Descrição sumario do script/modulo.

Mais info
Ainda mais info
"""

def nada():
    """Sumario da função.

    Descricao mais extensa e assinatura (tipo dos argumentos e do return
    nada: {} --> {}
    """

    return
```

## Projeto

### Testes unitários

PYTEST [HTTPS://DOCS.PYTEST.ORG/EN/7.1.X/](https://docs.pytest.org/en/7.1.x/)

```
def quadrado(x):  
    return x*x  
  
def test_1():  
    assert quadrado(2) == 4
```

DOCTEST [HTTPS://DOCS.PYTHON.ORG/3/LIBRARY/DOCTEST.HTML](https://docs.python.org/3/library/doctest.html)

```
def quadrado(x):  
    """Retorna o quadrado do seu argumento.  
  
    >>> quadrado(2)  
    4  
    >>> [quadrado(n) for n in range(10)]  
    [0, 1, 4, 9, 16, 25, 36, 49, 64, 81]  
    """  
    return x*x
```

In [ ]:

## Tarefas para esta semana

- Nas próximas aulas teóricas: Tipos Abstratos de Dados e introdução à programação funcional
  - Leituras recomendadas: Capítulo 9 (TADs)
- Nas aulas laboratoriais desta semana:
  - Avaliação: Cadeias de caracteres e listas
  - Primeira aula (L09): Dicionários
  - Segunda aula (L10): TADs
- **PROJETO!!!!**



In [ ]: