

---

# Programação em Python

## Tópicos

- Leitura e escrita de ficheiros. Exceções.

## Exercícios

1. Crie um ficheiro de texto com o conteúdo que desejar, sendo que deve conter mais do que 2 linhas. Grave o ficheiro com o nome `aula8.txt`.
  - a) Escreva um programa Python que leia e imprima o conteúdo do ficheiro de texto que definiu.
  - b) Escreva a função `word_count` que recebe como entrada o nome do ficheiro e calcula e apresenta o número de palavras no ficheiro. Garanta a utilização da declaração `with`.

*Sugestão: Use o método `.split()` para separar o conteúdo de uma string numa lista.*
  - c) Escreva a função `longest_word` que recebe como entrada o nome do ficheiro e retorna a palavra mais comprida do ficheiro.
2. O ficheiro `drawing.txt` contém linhas com as palavras `UP`, `DOWN` ou com um par de números. `UP` e `DOWN` são instruções para a tartaruga levantar ou pousar o pincel. Cada par representa um ponto  $(x, y)$  para onde a tartaruga deve deslocar-se (usando o método já definido `.goto`). Complete o programa `turtledraw.py` para ler as instruções do ficheiro e usar uma tartaruga para traçar o desenho.

*Sugestão: use o método `.split()` para dividir as linhas que têm coordenadas. (Adaptado do exercício 11.5 do livro [1].)*
3. Execute o programa `numInput.py` e experimente introduzir uma resposta não numérica. O programa terminará com uma exceção `ValueError` gerada pela chamada à função `float` dentro da função `floatInput`.
  - a) Altere a função `floatInput` para validar o valor: pede um valor, tenta convertê-lo em `float` e, se falhar, avisa o utilizador e repete.
  - b) Acrescente dois argumentos `min` e `max` e valide também se o valor está no intervalo  $[\text{min}, \text{max}]$ . Se não estiver, a função deve avisar e repetir.
  - c) \*Acrescente à função uma instrução `assert` para exigir que o argumento `min` não exceda `max`.