Programação II

Trabalho 1

Frequências de palavras num texto

Entrega a 1 de março de 2021

Este trabalho tem como objectivo a escrita de uma função Python que encontra as frequências de certas palavras no corpo de um texto e anota as linhas em que as palavras ocorrem.

Esta funcionalidade deve ser implementada através de uma função chamada `encontra_ocorrencias` que recebe o nome de dois ficheiros. O primeiro ficheiro contém o texto a analisar. O segundo ficheiro contém as palavras de interesse. Para todos os efeitos as palavras em cada linha devem ser separadas através do método `split()` da classe `String`.

A função `encontra_ocorrencias` devolve um dicionário. As chaves são as palavras que se encontram no ficheiro das palavras. Os valores são pares onde o primeiro elemento é o número de vezes que a palavra aparece no texto e o segundo elemento é o conjunto dos números das linhas onde a palavra aparece. A primeira linha do ficheiro do texto é identificada com o número 1.

Por exemplo, se a palavra `estrangeiro` constar no ficheiro das palavras e além disso aparecer, no ficheiro do texto, uma vez nas linhas 17 e 24, duas vezes na linha 31 e duas vezes na linha 93 (num total de 6 ocorrências), o dicionário resultante deve conter uma entrada da seguinte forma (lembre-se que a linguagem Python não imprime os conjuntos de forma ordenada).

```
'estrangeiro' : (6, {17,93,24,31})
```

As regras de boas práticas de desenvolvimento de software apontam para um número máximo de cerca de 10 linhas por função. Identifique as abstrações relevantes e implemente cada uma numa função separada.

Uma abstração importante é a função que lê todas as palavras constantes num ficheiro e devolve-as em forma de conjunto. O seu trabalho tem de incluir uma função `ler_palavras(nome_ficheiro)`. Por exemplo, para o ficheiro `palavras.txt` disponibilizado com este enunciado e que contém duas ocorrências da palavra `criptografia`, a chamada `ler_palavras("palavras.txt")` deve devolver um conjunto da seguinte forma.

```
{'algoritmo', 'artificial', 'cientista',
'criptografia', 'governo', 'inteligência'}
```

Para testar o seu código, estão disponíveis no Moodle dois ficheiros de texto, `turing.txt` e `palavras.txt`. Exemplo de execução:

```
>>> print(encontrar_palavras("turing.txt", "palavras.
    txt"))
{'governo': (10, {164, 166, 7, 172, 148, 150, 152,
    154, 156, 189}),
 'inteligência': (5, {1, 3, 135, 115, 119}),
 'cientista': (1, {1}),
 'criptografia': (1, {135}),
 'algoritmo': (2, {1, 117}),
 'artificial': (1, {115})}
```

Assuma que ambos os ficheiros existem e que podem ser lidos com sucesso. Além disso, assuma que qualquer um dos dois ficheiros podem estar vazios ou conter palavras repetidas. Para criar um conjunto vazio utilize a função `set()`.

Tome em especial atenção os seguintes pontos.

- Cada função que escrever (as obrigatórias `encontra_ocorrencias` e `ler_palavras`, bem como aquelas que porventura achar relevantes) deve estar equipada com uma descrição em formato `docstring`, tal como sugerido nas aulas.
- Não se esqueça de incluir o seu nome e número de estudante no início do ficheiro: `__author__ = Maria Lopes, 45638`.
- O vosso código será testado por um processo automatizado. É indispensável que este contenha duas funções com os nomes `encontra_ocorrencias` e `ler_palavras`, que as funções esperem exatamente o número e o tipo dos parâmetros e que devolvam exatamente as estruturas de dados enunciadas acima.

- Este é um trabalho de resolução individual. Os trabalhos devem ser entregues no Moodle até às 23:59 do dia 1 de março de 2021.
- Os trabalhos de todos os alunos serão comparados por uma aplicação de deteção de plágio em programas. Recorde o seguinte texto na secção Integridade Académica da Sinopse:

  "Alunos detetados em situação de fraude ou plágio (plagiadores e plagiados) em alguma prova ficam reprovados à disciplina e serão alvo de processo disciplinar, o que levará a um registo dessa incidência no processo de aluno, podendo conduzir à suspensão letiva."