

Aula Prática 7

Instruções para Submissão

Na aula prática de hoje, você terá que elaborar programas para resolver problemas diversos, conforme descrito abaixo. Cada uma das soluções deverá ser implementada em seu próprio arquivo com extensão `.py`. Por exemplo, a solução para o problema 1 deverá ser implementada em um arquivo chamado `problema1.py`, a solução para o problema 2 deverá ser implementada no arquivo `problema2.py` e assim por diante. Finalmente, submeta cada um dos arquivos pelo Moodle.

Dica: se você tiver problemas com caracteres especiais (caracteres com acentos, por exemplo), adicione a linha abaixo na primeira linha de todos os arquivos `*.py`

```
# -*- coding: utf-8 -*-
```

Problema 1

Escreva um programa que lê um arquivo chamado `texto.txt` e imprime na tela a linha do arquivo com o maior número de caracteres e a quantidade de caracteres dessa linha.

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo de execução do programa:

Assumindo que o arquivo `texto.txt` possui o conteúdo abaixo:

```
Oi
Tudo bem?
Ah, está bem mais ou menos!
O que aconteceu?
```

Seu programa deve imprimir:

Ah, está bem mais ou menos!
28

Problema 2

Escreva um programa que lê um arquivo chamado `texto.txt` e imprime na tela a palavra do arquivo com o maior número de caracteres e a quantidade de caracteres dessa palavra.

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo de execução do programa:

Assumindo que o arquivo `texto.txt` possui o conteúdo abaixo:

```
Oi
Tudo bem?
Ah, está bem mais ou menos!
O que aconteceu?
```

Seu programa deve imprimir:

aconteceu?
10

Problema 3

Escreva um programa que pede ao usuário para digitar um inteiro n , em seguida lê um arquivo chamado `texto.txt` e então imprime na tela todas as palavras do arquivo com um número de caracteres maior ou igual a n , mantendo a ordem com que as palavras aparecem no texto.

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo de execução do programa:

Assumindo que o arquivo `texto.txt` possui o conteúdo abaixo:

```
Oi
Tudo bem?
Ah, está bem mais ou menos!
O que aconteceu?
```

A execução do seu programa deve ser como exemplificado abaixo:

Digite um inteiro: 4

Tudo

bem?

está

mais

menos!

aconteceu?

Problema 4

Escreva um programa que lê um arquivo chamado `datas.txt` que contém em cada linha uma data no formato DD/MM/AAAA e então imprime na tela a data mais recente.

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo de execução do programa:

Assumindo que o arquivo `datas.txt` possui o conteúdo abaixo:

21/08/1987

04/10/1985

05/01/1987

06/10/1985

Seu programa deve imprimir:

21/08/1987

Problema 5

Escreva um programa que lê um arquivo chamado `notas.txt` que contém em cada linha o nome de um aluno e quatro notas entre 0 e 100. O programa deve então exibir na tela o nome e a média de todos os alunos cuja média seja maior ou igual a 60.

Observação: as mensagens exibidas para o usuário deverão ser exatamente como apresentado abaixo (mensagens exibidas com os comandos `input()` e `print()`).

Exemplo de execução do programa:

Assumindo que o arquivo `notas.txt` possui o conteúdo abaixo:

Leo 45 50 38 40

Lua 98 100 95 100

Bob 31 60 49 40
Ted 60 55 75 78

Seu programa deve imprimir:

Nome: Lua - Média: 98.25

Nome: Ted - Média: 67.00