

PROVA PRÁTICA 2

Leia todo o enunciado antes de começar a solução.

Considere os arquivos '**dados.txt**' e '**modelo.txt**', descritos a seguir.

O arquivo '**dados.txt**' contém valores em ponto flutuante (parte decimal separada por ponto), no formato tabular. O número de linhas é o mesmo em todas as colunas, porém a quantidade de colunas e linhas é desconhecida. Os dados contidos em cada linha são separados por um caractere de espaço. Não há comentários nem cabeçalho.

O arquivo '**modelo.txt**' contém 1 coluna de valores em ponto flutuante (parte decimal separada por ponto), com um número arbitrário de linhas. Não há comentários nem cabeçalho.

Elabore um programa em Python que execute as seguintes tarefas:

- a) Emitir um alerta na tela no caso de haver **valores repetidos em alguma coluna**. Use uma função (definida no programa) para executar tal verificação. A função deve retornar somente True ou False.
- b) Ordenar as colunas (em ordem crescente) do arquivo '**dados.txt**' e salvar o resultado no arquivo '**dados_ordenados.txt**'.
- c) Seja n o menor valor entre o número de linhas nos arquivos '**dados_ordenados.txt**' e '**modelo.txt**'. Gerar um novo arquivo chamado '**saida.txt**' contendo **2 colunas separadas por espaço**, com as seguintes informações:
 - i) primeira coluna: a i -ésima linha deve conter a **média aritmética** dos valores contidos na i -ésima linha do arquivo '**dados_ordenados.txt**', com 6 casas decimais, onde i vai de 1 a n ;
 - ii) segunda coluna: a i -ésima linha deve conter o valor do **erro** entre a média definida acima e o valor da i -ésima linha do arquivo '**modelo.txt**', **relativo à média**, com 6 casas decimais, onde i vai de 1 a n . Use uma função (definida no programa) para calcular o erro. A função deve retornar o valor individual ou de todas as linhas. Observação: O erro entre x e y , relativo a x , é dado por $|x - y| / |x|$.

Regras de implementação e considerações úteis:

- 1. O programa deve implementar tratamento de erro na abertura de arquivos. O programa deve escrever uma mensagem na tela em caso de erro, **indicando o nome do arquivo com problema**.
- 2. Os arquivos de entrada não devem ser alterados (nem o nome nem o conteúdo).
- 3. Considere que os arquivos de entrada possuem a formatação correta de acordo com a definição do enunciado.
- 4. O programa deve gerar somente o arquivo de saída solicitado, **mantendo o nome e a extensão**.
- 5. Considere que os arquivos manipulados encontram-se no mesmo diretório de execução do código.

6. Apenas bibliotecas padronizadas da linguagem são permitidas.
7. O programa não deve solicitar nenhuma digitação ao usuário.
8. O que não estiver explicitado no enunciado é de escolha do programador.

Pontuação da prova	
Leitura dos arquivos de entrada	1,5 ponto
Verificação (função e uso)	2,0 pontos
Ordenação e salvar dados ordenados	1,5 ponto
Cálculo da média	1,5 ponto
Cálculo do erro (função e resultado)	1,5 ponto
Escrita do arquivo saida.txt	2,0 pontos