

 <b>INSTITUTO FEDERAL</b> Espírito Santo Campus Serra	<b>Curso de Engenharia de Controle e Automação</b>			
<b>Componente Curricular:</b> Programação Orientada a Objetos. <b>Professor:</b> Hilário Tomaz Alves de Oliveira				
	<b>Semestre:</b> 2022.1	<b>Período:</b> 3º	<b>Turma:</b> Noite	<b>Data de Entrega:</b> 29/05/2022

## Lista de Exercícios 3 – Manipulação de Arquivos

### Observações:

- A solução de cada questão que necessita de implementação de código deve estar contida em um arquivo de código na linguagem Python (extensão .py). Para padronizar utilize o seguinte padrão de nomenclatura.
  - l#NumeroListaq#NumeroQuestao.py
    - Nos quais:
      - #NumeroLista deve ser trocado pelo número da lista;
      - #NumeroQuestao deve ser trocado pelo número da questão.
    - **Exemplos:** l1q1.py, l3q4.py, l5q10.py, e assim sucessivamente
  - Ao final cada aluno deve enviar uma pasta compactada contendo todos os arquivos com suas soluções.
  - Nota: 7,00

**Questão 1.** Desenvolva um programa em Python que solicite ao usuário o nome de um arquivo existente e uma palavra. Seu programa deve retornar o número de vezes que a palavra lida aparece no arquivo.

**Questão 2.** Escreva um programa na linguagem Python que leia um arquivo chamado de **nomes.txt** contendo vários nomes de pessoas (1 por linha). Seu programa deve ordenar os nomes e gerar um novo arquivo chamado de **nomes\_ordenados.txt** com os nomes

ordenados. **DICA:** Leias os nomes dos arquivos e armazene em uma lista de nomes. Pesquise na internet como ordenar uma lista em Python.

**Questão 3.** Faça um programa na linguagem Python que apure o resultado de uma votação para determinar o personagem do desenho “Dragon Ball” favorito. Suponha que existam 5 candidatos cujos códigos de identificação são: 1- Goku, 2 - Vegeta, 3 - Gohan, 4 - Broly, 5 - Freeza. Considere que existe um arquivo texto chamado de “votos.txt” que contém, em cada linha, um determinado voto (um voto é representado pelo código de identificação do candidato). Seu programa deverá ler o arquivo de votos e apresentar, como resultado, o nome do candidato e a quantidade de votos do candidato mais votado, o código de identificação e a quantidade de votos do candidato menos votado, e a quantidade de votos nulos (um voto nulo é um voto cujo código de identificação é um inteiro diferente de 1, 2, 3, 4 e 5).

**Questão 4.** Desenvolva na linguagem Python uma função que recebe por parâmetro dois nomes de arquivos e copia o conteúdo do primeiro arquivo para o segundo arquivo. Considere que o conteúdo do arquivo de origem é um texto. Sua função NÃO deve copiar linhas comentadas (que começam com #). Teste sua função com diferentes arquivos de entrada.

**Questão 5.** Desenvolva um programa na linguagem Python que leia do usuário o caminho de um diretório (pasta). Seu programa deve ler e imprimir na tela todos os arquivos e pastas contidos no diretório informado pelo usuário.

**Questão 6.** Implemente um programa na linguagem Python que analise um arquivo chamado de “pontuacoes.csv” contendo dados das pontuações dos jogadores nas partidas. Seu programa deve ler, processar o arquivo e gerar um outro arquivo CSV chamado “maiores\_pontuacoes.csv” em que cada linha do arquivo tem o nome do jogador e sua maior pontuações. Seguem exemplos dos arquivos “pontuacoes.csv” e “maiores\_pontuacoes.csv”:

- pontuacoes.csv

```
nome;pontuações
João da Silva;23
Maria Alice;27
João da Silva;12
Maria Alice;26
João da Silva;26
Ana Clara;10
Ana Clara;40
```

- maiores\_pontuacoes.csv

```
nome;maior pontuação
Ana Clara;40
João da Silva;26
Maria Alice;27
```

**Questão 7.** Implemente um programa na linguagem Python contendo uma função que recebe como argumentos o caminho de dois arquivos. O primeiro arquivo contém nomes de alunos e o segundo arquivo contém as notas dos alunos. No primeiro arquivo, cada linha corresponde ao nome de um aluno e no segundo arquivo, cada linha corresponde às notas dos alunos (uma ou mais). Por exemplo, a primeira linha do primeiro arquivo tem o nome do aluno e suas notas estão na primeira linha do segundo arquivo e assim sucessivamente. As notas de cada aluno estão separadas umas das outras por espaços em branco. Leia os dois arquivos e gere um terceiro arquivo que contém o nome do aluno seguido da média de suas notas.

**Questão 8.** Uma empresa está tendo problemas de espaço em disco no seu servidor de arquivos. Para tentar resolver este problema, o administrador de rede precisa saber qual o espaço ocupado pelos usuários, e identificar os usuários com maior espaço ocupado. Usando um programa, ele conseguiu gerar o seguinte arquivo, chamado "usuarios.csv":

```
Nome;Espaço
Alexandre;456123789
Anderson;1245698456
Antonio;123456456
Carlos;91257581
```

Cesar;987458

Rosemary;789456125

O arquivo CSV gerado possui duas colunas. A primeira coluna tem o nome do usuário e a segunda tem o espaço em disco que o usuário ocupa em bytes. Desenvolva um programa em Python que leia o arquivo "usuarios.csv" e gera um outro arquivo chamado de "relatorio.txt" contendo um relatório do uso de disco do servidor. Seu relatório deve ter o seguinte formato:

Usuário	Espaço utilizado	% do uso
1 alexandre	434,99 MB	16,85%
2 anderson	1187,99 MB	46,02%
3 antonio	117,73 MB	4,56%
4 carlos	87,03 MB	3,37%
5 cesar	0,94 MB	0,04%
6 rosemary	752,88 MB	29,16%

Espaço total ocupado: 2581,57 MB

Espaço médio ocupado: 430,26 MB

**OBS:** Observe que no relatório a ser gerado o espaço de disco está sendo representado em megabytes. Logo, crie uma função que realiza a conversão de bytes para megabytes.

**Questão 9.** Implemente um programa na linguagem Python que crie e manipule um arquivo (binário) de produtos. Para cada produto, seu sistema deve solicitar os seguintes dados: **código, nome, quantidade em estoque e preço**. Seu programa deverá exibir o seguinte menu:

#### CADASTRO DE PRODUTOS

- 1 - INSERIR UM NOVO PRODUTO
- 2 - ALTERAR UM PRODUTO
- 3 - EXCLUIR UM PRODUTO
- 4 - EXCLUIR TODOS OS PRODUTOS
- 5 - CONSULTAR UM PRODUTO
- 6 - LISTAR TODOS OS PRODUTOS

7 - EXIBIR DADOS ESTATÍSTICOS

8 - SAIR

DIGITE SUA OPÇÃO:

### **OBSERVAÇÕES:**

- Na opção 1 e 2, seu programa não deve permitir o cadastro de dois ou mais produtos com o mesmo código.
- Nas opções 3 e 5 você deve usar o código para encontrar o produto desejado.
- Pesquise estratégias que você pode utilizar para realizar as opções 2 e 3.
- Na opção 7 deverá ser apresentada as seguintes estatísticas:
  - a) A quantidade de produtos cadastrados;
  - b) A quantidade e o percentual de produtos com estoque baixo ( $\leq 20$ );
  - c) A quantidade e o percentual de produtos acima da média;
  - d) A média geral e o desvio padrão dos preços dos produtos cadastrados.
- Modularize seu programa criando diferentes funções para realizar as tarefas necessárias.