



## TRABALHO DE ESTRUTURA DE DADOS II

ORDENAÇÃO DE DADOS  
Peso 3 da N2



### OBJETIVO

Desenvolver um programa que implemente um dos algoritmos de ordenação (MERGE, HEAP, RADIX ou SHELL). O algoritmo será sorteado por grupo.

### ESPECIFICAÇÕES DO PROGRAMA

O programa deverá ler um arquivo de entrada do tipo `.TXT`, chamado `ENTRADA.TXT`, com o seguinte formato:

CODIGO tab DESCRIÇÃO tab UNIDADE tab VALOR
--

### DEFINIÇÕES DOS CAMPOS

- CODIGO: Inteiro positivo sem casa decimal.
- DESCRIÇÃO: String com tamanho mínimo de 1 caractere.
- UNIDADE: String com tamanho mínimo de 1 caractere.
- VALOR: Float positivo, maior ou igual a zero.

#### Observações:

- Os campos UNIDADE e VALOR podem ter valores duplicados.
- Nenhum campo estará vazio.

### FUNCIONALIDADES DO PROGRAMA

#### 1. Leitura do Arquivo:

- O programa deve ler o arquivo `ENTRADA.TXT` conforme o formato especificado.

#### 2. Escolha do Campo e Sentido de Ordenação:

- O programa deve permitir ao usuário escolher por qual dos quatro campos (CODIGO, DESCRIÇÃO, UNIDADE, VALOR) a ordenação será feita.
- O programa deve permitir ao usuário escolher o sentido da ordenação: Ascendente ou Descendente.

#### 3. Ordenação Estável:

- Caso o campo escolhido para ordenação possua valores duplicados, o algoritmo deve manter a ordem original do arquivo de entrada, garantindo a estabilidade da ordenação (ALGORITMO ESTÁVEL).

#### 4. Saída do Programa:

- O resultado da ordenação deve ser gravado em um arquivo de saída no formato `.TXT`, com o mesmo *layout* do arquivo de entrada:

CODIGO tab DESCRIÇÃO tab UNIDADE tab VALOR
--

- O nome do arquivo será composto pelos nomes dos alunos separados por um *underline* (`_`) e terá a extensão `.TXT`.

### IMPLEMENTAÇÃO

- O programa pode ser desenvolvido nas linguagens C/C++, Java ou Python.

## **ENTREGA**

- O trabalho pode ser realizado individualmente ou em duplas.
- O nome do projeto e da pasta deve ser o nome dos alunos separados por um *underline* (`\_`).
- O aluno deve enviar todo o projeto (todos os arquivos que fazem parte dele) funcional (ou seja, funcionando) compactado em um arquivo `.ZIP`.
- O nome do arquivo `.ZIP` deve ser o nome dos alunos separados por um *underline*.
- Os alunos deverão trazer seu computador pessoal para apresentação.

## **INSTRUÇÕES DE ENVIO**

- Enviar o arquivo `.ZIP` por e-mail até as 17h do dia 24/06/2024 para [evandro.viapiana@uniavan.edu.br](mailto:evandro.viapiana@uniavan.edu.br) com o assunto composto pelos nomes dos alunos separados por um *underline*.

## **CRITÉRIOS DE AVALIAÇÃO**

- Funcionamento do programa.
- Adesão às especificações de ordenação e estabilidade.
- Cumprimento das regras de nomenclatura.
- Entrega dentro do prazo estipulado.

## **APRESENTAÇÃO**

- Os alunos farão a execução e apresentação do trabalho no seu próprio computador.
- No dia da apresentação, 24/06/2024, o professor poderá pedir explicações sobre o código fonte e solicitar alterações que deverão ser feitas na hora.
- Caso não consigam realizar as alterações solicitadas, haverá redução de nota.