

# Trabalho Final – Ordenação Externa com Multiway Merge Sort

Disciplina: Algoritmos e Estruturas de Dados II  
Profa. Dra. Simone Dominico

Entrega: até 24/06/2025  
Defesa: Aula de 25 e 27/06/2025

## Objetivo

Neste trabalho, você aplicará os conhecimentos de ordenação eficiente, estruturas de dados e acesso a arquivos para resolver um problema real: ordenar dados que não cabem na memória RAM. O foco está em compreender e aplicar o algoritmo de ordenação externa conhecido como *Multiway Merge Sort*.

Você precisará aprender a:

- Dividir dados em partes manipuláveis na memória (*runs*);
- Ordenar e armazenar essas partes em arquivos temporários;
- Utilizar *min-heap* para intercalar os dados;
- Gerar um arquivo final ordenado, mesmo com restrição de memória.

## Descrição do Problema

Você tem um arquivo com centenas de milhares de números, mas só consegue carregar 10 mil por vez na memória. Como ordenar esse arquivo?

O algoritmo funcionará em duas etapas:

### Etapa 1 – Geração de Runs

1. Divida o arquivo grande em blocos menores (ex: 10.000 linhas).
2. Ordene cada bloco em memória.
3. Salve esses blocos ordenados como arquivos temporários (`run1.txt`, `run2.txt`, ...).

## Etapa 2 – Intercalação k-way

1. Reabra os arquivos temporários.
2. Ordene os arquivos - escolha a melhor estrutura entre as discutidas em aula.
3. Escreva os elementos ordenadamente em um novo arquivo (`saida.txt`).

## Entrega

### 1. Código-fonte comentado

A linguagem pode ser em Python, C, C++, preferível em C.

### 2. Relatório (até 3 páginas)

- Explique como dividiu e ordenou os dados;
- Mostre o funcionamento da estrutura durante a intercalação;
- Analise como o número de arquivos  $k$  influencia o desempenho;
- Liste as dificuldades encontradas e como resolveu.

### 3. Apresentação (obrigatória)

- Duração: 8 minutos;
- Mostre que compreendeu os conceitos envolvidos.

## Organização

- Trabalho em dupla ou individual;
- Ambos devem apresentar na defesa.

## Avaliação

Critério	Pontos
Clareza e domínio na apresentação	20
Código funcional e bem estruturado	5
Relatório técnico claro	3
Testes e análise do impacto de $k$	2
<b>Total</b>	<b>30</b>

## Dicas importantes

- Comece com arquivos pequenos (ex: 1.000 números);
- Teste com  $k = 2, 3, 4$  arquivos temporários;
- Divida o trabalho: leitura  $\rightarrow$  ordenação  $\rightarrow$  gravação  $\rightarrow$  intercalação.

**Os dados de entrada estão disponíveis no Ambiente Virtual, a saída deve ser no mesmo formato em ordem crescente.**