

# Trabalho Grafos Parte 2

## Da Entrega

Deve ser enviado **pelo classroom da turma** em um único arquivo .zip até dia **11/02/2025** com a seguinte estrutura de pasta e arquivos (**e nomes dos arquivos**):

TrabalhoGrafosGrupoX.zip

```
| docs/
| | documentacao.pdf
| | apresentacao.pdf
| | contribuidores.pdf      (o que cada membro fez, junto com o link do github)
| | descricao.pdf          (este documento)
| include/
| | *.h
| | *.hpp
| | *.tpp
| src/
| | *.c
| | *.cpp
| entradas/
| | *.txt
| main.cpp
| README      (explicação simples e considerações sobre a execução do programa)
```

## Da apresentação

A apresentação será feita em sala de aula no esquema de elevator pitch onde cada aluno terá exatamente 1 minuto para apresentar as atividades que ele executou no projeto. Após isso, terá um periodo de perguntas por parte do professor.

Não será necessário que todos os alunos assistam todas as apresentações, mas todas serão livres para quem quiser assistir. O horário de apresentação de cada grupo dentro do periodo da aula será disponibilizado posteriormente.

# Da avaliação

O trabalho será avaliado em 100 pontos e depois será ponderado conforme seu valor na nota final da disciplina.

O calculo da nota será dado por:  $\sqrt{T * \sqrt{A * P}}$  onde A, T e P valem 100 pontos e representam:

- A - Apresentação
- T - Avaliação Geral do trabalho
- P - Participação do aluno no desenvolvimento do código

# Do código

- Adicionar as funções novo\_no, nova\_aresta, deleta\_no e deleta\_aresta nas estruturas grafo\_matriz e grafo\_lista.
  - Para o grafo\_matriz, a matriz agora deve ser alocada de forma dinâmica começando com o tamanho 10 e sendo reconstruída com o dobro do tamanho sempre que ela não tiver capacidade de incluir um novo nó.
  - Além disso, quando um nó for removido no grafo\_matriz, os ids devem ser recalculados, gerando assim um grafo isomorfo ao original.
- Adicionar uma função que calcula a menor distância entre 2 nós.

# Da execução

Após compilado, o código deve ser executado via terminal com as seguintes linhas de comando:

**Caso 1:** `main.out -d -m grafo.txt`

- Imprime descrição do grafo após carregá-lo com matriz de adjacência e excluir o nó de id 1 (se existente) e a primeira aresta do nó de id 2 (se existente) como no exemplo abaixo:

grafo.txt

**Excluindo nó 1...**

**Excluindo primeira aresta do nó 2...**

Grau: 3

Ordem: 3

Direcionado: Sim

Vertices ponderados: Sim

Arestas ponderadas: Sim

Completo: Sim

**Maior menor distância: (3-9) 8.23**

Maior menor distância possui o id dos nós mais distantes entre si e seu peso.

**Caso 2:** `main.out -d -l grafo.txt`

- Mesma coisa que o caso 1, mas carregando o grafo com lista encadeada.