



**Pontifícia Universidade Católica de Minas Gerais**  
**Instituto de Ciências Exatas e Informática**  
**Algoritmos e Estruturas de Dados III**

## **Trabalho Prático I**

O trabalho deve ser feito em grupos de no **máximo 2 alunos**

**Data de entrega:** 20/03/2025

**Valor:** 5 pontos

### **Etapas dos TPS:**

- **Etapa 1: Criação da base de dados: Implementação**
- Etapa 2: Indexação: Implementação → 24/04
- Etapa 3: Compactação: Implementação → 08/05
- Etapa 4: Casamento de Padrões: Implementação → 22/05
- Etapa 5: Criptografia: Implementação → 18/05

### **Descrição do TP1:**

O primeiro trabalho prático da nossa disciplina é a implementação de um CRUD genérico, isto é, de uma classe que faça inclusões (*create*), leituras (*read*), alterações (*update*) e exclusões (*delete*) em um arquivo. Como queremos que ele seja genérico, então essa classe deve ser capaz de escrever registros de qualquer entidade (ex.: produtos, clientes, usuários, livros, eventos, mensagens, etc.).

Você deverá selecionar uma base de dados que permita a representação das entidades em registros com campos de cada um dos tipos abaixo (no mínimo):

1. ID do registro
2. String de tamanho variável
3. Data
4. Lista de valores String com indicador de quantidade
5. Tipo numérico

Fique à vontade para selecionar uma base de dados disponível em fontes de dados como o [Kaggle](#), [UCI Datasets](#) ou outra que desejar, desde que cumpra os requisitos acima e seja de domínio público.

Após selecionar a base de dados desejada, você deverá implementar um sistema responsável por realizar operações de CRUD (create, read, update e delete) em um arquivo sequencial.

### **Orientações:**

- O sistema deve ser implementado em C Like. Todo o código deve ser de autoria do grupo (com exceção para bibliotecas/classes relacionadas a aberturas e escritas/leituras de arquivos e conversões entre atributos e campos).



**Pontifícia Universidade Católica de Minas Gerais**  
**Instituto de Ciências Exatas e Informática**  
**Algoritmos e Estruturas de Dados III**

- Caso algum dos tipos acima citados não esteja presente na base de dados escolhida, você poderá estabelecer algum critério que permita a inclusão do campo. Exemplo: inclusão de ID, inclusão da data/hora de carga.
- O código deve ser comentado de modo a se compreender a lógica utilizada.
- A estrutura do arquivo deve ser a seguinte:
  - Deve-se utilizar um int no cabeçalho para armazenar o último valor de id utilizado.
  - Os registros do arquivo devem ser compostos por:
    - Lápide - Byte que indica se o registro é válido ou se é um registro excluído;
    - Indicador de tamanho do registro - Número inteiro que indica o tamanho do vetor de bytes;
    - Vetor de bytes - Bytes que descrevem o objeto.
- Os objetos utilizados devem possuir os atributos que representam a entidade da base de dados que você escolheu.

### **Parte 1 - CRUD:**

O sistema deverá oferecer uma tela inicial (com uso pelo terminal) com um menu com as seguintes opções:

- Realizar a carga da base de dados selecionada, através da importação de arquivo CSV, de rota de API ou outro formato que julgar pertinente.
- Criar um registro (objeto) -> esse método deve receber um objeto como parâmetro e criar o registro referente a esse objeto no arquivo.
- Ler um registro (id) -> essa função deve receber um id como parâmetro, percorrer o arquivo e retornar os dados do id informado.
- Ler um conjunto (id) -> essa função deve receber um conjunto de ids como parâmetro, percorrer o arquivo e retornar os dados de cada id informado.
- Atualizar um registro -> esse método deve receber novas informações sobre um objeto e atualizar os valores dele no arquivo.
- Deletar um registro (id) -> esse método deve receber um id como parâmetro, percorrer o arquivo e colocar uma marcação (lápide) no registro que será considerado deletado.

### **O que deve ser entregue:**

#### **Implementação**

- Carga da Base de Dados;
- CRUD;



**Pontifícia Universidade Católica de Minas Gerais**  
**Instituto de Ciências Exatas e Informática**  
**Algoritmos e Estruturas de Dados III**

Além da Implementação, o grupo deverá apresentar, máximo 5 minutos, o trabalho, com:

- Explicação das principais decisões de implementação dos códigos criados.
- Demonstração da execução do sistema.
- Testes e resultados realizados

**Critérios para avaliação**

Critério	Atende perfeitamente (1,0)	Atende de forma satisfatória (0,8)	Atende parcialmente (0,6)	Atende de forma insatisfatória (0,3)	Não atende (0)
<b>Correção e robustez dos programas</b>	O sistema está livre de erros, falhas e lida bem com entradas inesperadas.	O sistema tem poucos erros ou falhas, mas lida corretamente com entradas esperadas e alguns edge cases.	O sistema falha em algumas condições ou não lida com todas as entradas inesperadas corretamente.	O sistema tem falhas que afetam o funcionamento básico e não lida bem com entradas inesperadas.	O sistema apresenta falhas graves que comprometem a execução e não lida com entradas adequadas.
<b>Conformidade às especificações</b>	A implementação segue fielmente todas as especificações do problema, sem falhas.	A implementação segue a maioria das especificações, com algumas pequenas variações ou omissões.	A implementação segue parcialmente as especificações, mas omite ou erra em pontos importantes.	A implementação falha em aspectos importantes das especificações, comprometendo a funcionalidade.	A implementação não segue as especificações de forma significativa, prejudicando a solução.
<b>Clareza de codificação</b>	O código é bem estruturado, com boa nomenclatura de variáveis, funções e comentários que facilitam o entendimento.	O código é estruturado de forma geral, mas poderia ser mais claro com alguns ajustes.	O código é razoável, mas possui áreas que poderiam ser mais legíveis ou organizadas.	O código é difícil de entender devido a falta de clareza ou nomeações inadequadas em algumas partes.	O código é confuso, mal estruturado e difícil de entender sem uma análise profunda.
<b>Critérios de escolha</b>	As escolhas de técnicas e soluções são bem fundamentadas, apropriadas e justificadas de forma clara.	As escolhas de técnicas são geralmente apropriadas, mas com algumas escolhas questionáveis.	Algumas escolhas não são as mais apropriadas, ou não há uma justificativa clara para as decisões.	As escolhas feitas não são as mais adequadas ou são pouco justificadas.	As escolhas de técnicas são inadequadas ou mal fundamentadas para o problema.
<b>2. Apresentação (01 ponto)</b>					
Critério	Atende perfeitamente (1,0)	Atende de forma satisfatória (0,8)	Atende parcialmente (0,6)	Atende de forma insatisfatória (0,3)	Não atende (0)
<b>Apresentação do trabalho</b>	A apresentação é clara, bem estruturada, com explicações detalhadas sobre a implementação e desafios enfrentados.	A apresentação está boa, mas poderia ser mais detalhada ou clara em alguns pontos.	A apresentação tem pontos importantes faltando ou não está completamente clara.	A apresentação é difícil de seguir e carece de explicações suficientes sobre a implementação.	A apresentação é confusa, sem uma estrutura clara ou explicações suficientes.