



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

Trabalho Prático II
5 pontos

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

Data de entrega: 12/04/2025

Penalidade por atraso: a cada dia corrido de atraso, a nota será penalizada em 2 pontos.

Penalidade por cópia: trabalhos iguais não são aceitos (nota 0).

Etapas dos TPS:

- Etapa 1: Criação da base de dados, Manipulação de Arquivo Sequencial: Implementação + Vídeo
- **Etapa 2: Manipulação de Arquivo Indexado com Árvore B+, Hash e Lista Invertida: Implementação + Vídeo**
- Etapa 3: Compactação com Huffman e LZW: Implementação + Vídeo
- Etapa 3: Casamento de Padrões e Criptografia: Implementação + Relatório Final

Descrição do TP1:

Neste trabalho, você deverá manipular de forma indexada a base de dados criada no TP1.

INDEXAÇÃO:

- Orientações para a criação do arquivo de índices usando **Árvore B, B+ ou B***:
 - O arquivo de índices deve usar a estrutura de Árvore B, B+ ou B*, usando como chave o campo *id*.
 - Você deve identificar e escolher qual árvore será usada (B, B+ ou B*). A escolha deve ser justificada.
 - A ordem da árvore deve ser parametrizada.
 - O arquivo de índices deve conter o id e a posição do registro (referente a esse id) no arquivo de dados.
 - Sempre que acontecerem alterações no arquivo de dados, novas alterações devem ser feitas no arquivo de índices, mantendo sempre a coerência entre esses arquivos.
 - O arquivo de índice criado deve possibilitar a realização de buscas no arquivo de dados.
 - Explique no vídeo todas as decisões e escolhas feitas.



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

- Orientações para a criação do arquivo de índices usando **Hashing Estendido**:
 - O arquivo de índices deve usar a estrutura de Hashing Estendido, usando como chave o campo *id*.
 - Você deve identificar e escolher em seu arquivo o campo que será indexado. Cada escolha deve ser justificada..
 - Deve-se usar a função hash $h(k) = k \bmod 2^p$, em que p é o número de bits (profundidade) usado no diretório, sendo que cada bucket pode armazenar até X registros, sendo X 5% do tamanho inicial de sua base.
 - O arquivo de índices deve conter o *id* e a posição do registro (referente a esse *id*) no arquivo de dados.
 - Sempre que acontecerem alterações no arquivo de dados, novas alterações devem ser feitas no arquivo de índices, mantendo sempre a coerência entre esses arquivos.
 - Deve existir a possibilidade de realizar buscas usando a estrutura de índices de Hashing Estendido.
 - Explique no vídeo todas as decisões e escolhas feitas.

- Orientações sobre a criação da **lista invertida**.
 - *Deve-se criar dois arquivos contendo listas invertidas.*
 - Você deve identificar e escolher em seu arquivo como as listas invertidas serão aplicadas.
 - Explique no vídeo todas as decisões e escolhas feitas.
 - O sistema deverá realizar alterações nas listas invertidas sempre que novos registros forem inseridos, alterados ou deletados no arquivo de dados.
 - O sistema deve ser capaz de receber uma busca usando as listas invertidas criadas. Inclusive, deve ser possível utilizar as duas listas invertidas em uma mesma pesquisa

- Orientações sobre operações sobre Índices.
 - Realize as operações de CRUD agora com apoio dos índices.
 - A cada operação de CRUD você deverá indicar se a operação será feita utilizando Árvore B, Hash ou a Lista Invertida

O que deve ser entregue:

Implementação

- CRUD com apoio dos índices

Além da Implementação, o grupo deve criar um vídeo (duração máxima de 10 minutos), com:

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



Pontifícia Universidade Católica de Minas Gerais

Instituto de Ciências Exatas e Informática

Algoritmos e Estruturas de Dados III

Critérios para avaliação

- Implementação do sistema (5 pontos)
 - Correção e robustez dos programas
 - Conformidade às especificações
 - Clareza de codificação
 - Critérios de escolha
- Vídeo (1 ponto)

Observação final: ponto(s) extra(s) pode(m) ser dado(s) para trabalhos considerados excelentes.