



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: 22/05/2025

Valor: 5 pontos

Etapas dos TPS:

- Etapa 1: Criação da base de dados: Implementação
- Etapa 2: Indexação: Implementação → 08/05
- **Etapa 3: Compactação: Implementação → 22/05**
- Etapa 4: Casamento de Padrões: Implementação → 05/06
- Etapa 5: Criptografia: Implementação → 19/06

Descrição do TP1:

Neste trabalho, você deverá implementar Compressão de Dados dentro do contexto do seu Trabalho.

- Algoritmos de Compressão de Dados: Huffman e LZW.
- Orientações para a criação da Compressão de Dados:
 - No seu menu de opções apresentado ao usuário, ofereça a possibilidade de escolher uma opção para realizar a compressão na base de dados criada e crie uma opção para ele realizar a descompressão de alguma versão de compressão criada.
 - Quando o usuário escolher a opção de compressão, a base de dados deve passar pela compressão usando os dois algoritmos e os novos arquivos gerados devem seguir o nome: "nomeArquivoNomeAlgoritmoCompressaoX", em que X representa a versão da compressão, nomeArquivo o nome original do arquivo e nomeAlgoritmo o nome do algoritmo usado.
 - Além de realizar a compressão e gerar os novos arquivos, o algoritmo deve mostrar para o usuário a porcentagem de ganho ou perda de cada algoritmo e o tempo de execução de cada, comparando as execuções e mostrando qual algoritmo (Huffman ou LZW) foi melhor para aquela condição.
 - A compressão deve ser feita em todos os campos do arquivo, incluindo o cabeçalho, indicadores de tamanho de strings e afins.
 - Caso o usuário escolha descompactar o arquivo, ele deve passar a versão "X" que deseja, e a descompressão deve substituir o arquivo de dados pelo arquivo gerado pela descompressão. Novamente, o algoritmo deve mostrar para o usuário o tempo de execução de cada, comparando as execuções e mostrando qual algoritmo (Huffman ou LZW) foi melhor para aquela condição de descompactação.
 - Para o algoritmo de LZW, você é responsável pela definição do dicionário inicial.
 - As decisões relacionadas ao algoritmo são de responsabilidade do grupo.



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

O que deve ser entregue:

Implementação

- Códigos fontes.

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)
Correção e robustez dos programas	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.
Conformidade às especificações	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.
Clareza de codificação	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.
Critérios de escolha	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.
2. Apresentação (01 ponto)					
Critério	Atende perfeitamente (1,0)	Atende de forma satisfatória (0,8)	Atende parcialmente (0,6)	Atende de forma insatisfatória (0,3)	Não atende (0)
Apresentação do trabalho	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.