

Trabalho Individual: Implementação do Código de Huffman para Compressão de Texto

Michel Pires, Centro Federal de Educação Tecnológica de Minas Gerais

Data de Entrega: 06/12/2025

Valor: 10 pontos (Implementação + Repositório GitHub)

1 Apresentação Geral

O presente trabalho tem como objetivo consolidar o conhecimento sobre **estruturas em árvore** e **compressão de dados** por meio da implementação prática do **algoritmo de Huffman**. Cada aluno deverá desenvolver um programa capaz de realizar a compressão de pequenos trechos de texto utilizando o código de Huffman como método de codificação estatística.

O algoritmo de Huffman, amplamente utilizado em sistemas de compressão sem perdas (*lossless*), baseia-se na frequência de ocorrência dos símbolos em uma mensagem. A partir dessas frequências, é construída uma **árvore binária ponderada**, que associa códigos binários menores aos símbolos mais frequentes e códigos maiores aos menos frequentes, promovendo a redução do tamanho total da representação.

2 Descrição do Trabalho

Cada aluno deverá implementar um programa que:

1. Leia de um arquivo denominado `input.dat` um conjunto de pequenos textos. Cada texto deverá estar separado por uma linha em branco e conter frases em português com palavras repetidas com diferentes frequências;
2. Para cada texto lido, calcule a frequência de cada palavra e construa a **árvore de Huffman** correspondente;
3. Gere o código binário de cada símbolo, de acordo com o percurso da árvore (0 para ramo esquerdo, 1 para ramo direito);
4. Com base na árvore gerada, realize a compressão de cada trecho de texto, produzindo um arquivo de saída denominado `output.dat`;
5. O arquivo `output.dat` deverá conter, para cada texto processado:
 - A estrutura da árvore de Huffman (em formato textual ou serializado);
 - O conjunto dos códigos gerados para cada símbolo;
 - O texto comprimido correspondente;
 - Informações suficientes para permitir, se desejado, a **decodificação completa** do texto original.

3 Conjunto de Dados

O arquivo `input.dat` deverá conter pelo menos três trechos de texto de tamanho reduzido (duas a três frases cada). Recomenda-se que esses textos apresentem palavras recorrentes, como nos exemplos abaixo:

- “O computador executa instruções em alta velocidade e processa dados com precisão.”
- “A memória armazena informações que são acessadas rapidamente pela CPU.”
- “Os sistemas operacionais controlam os recursos e coordenam as tarefas do processador.”

Esses textos são apenas ilustrativos e podem ser substituídos por outros equivalentes, desde que mantenham a característica de recorrência de palavras, condição essencial para observar a eficiência do algoritmo de Huffman.

4 Requisitos de Implementação

- A linguagem de programação é de livre escolha, desde que o código seja compatível com sistemas **Linux (Ubuntu 24.04)** e de fácil compilação;
- O programa deve ser executável por linha de comando, sem dependências externas complexas;
- O código deve estar devidamente comentado, com funções e estruturas claramente identificadas;
- O programa deve gerar os arquivos `input.dat` e `output.dat` na pasta designada como **data**.

5 Entrega e Avaliação

A entrega deste trabalho é **individual** e deverá ser realizada até o dia **06/12/2025**, por meio do **AVA institucional**, contendo apenas o link para o repositório **Git** público que hospeda o código-fonte.

É de responsabilidade de cada aluno garantir que o link fornecido possa ser executado corretamente com o comando:

```
git clone <URL>
```

Repositórios inacessíveis, privados ou com dependências não declaradas acarretarão a perda da pontuação correspondente.

Critérios de Avaliação (10 pontos)

- Correção e completude da implementação — 5 pontos;
- Organização e clareza do código-fonte — 2 pontos;
- Geração correta dos arquivos `output.dat` — 2 pontos;
- Disponibilidade e reprodutibilidade do repositório Git — 1 ponto.

6 Considerações Finais

Este exercício tem por finalidade reforçar os conceitos de codificação eficiente de dados e manipulação de estruturas em árvore, estimulando a compreensão prática dos princípios de compressão sem perdas. O entendimento do funcionamento do código de Huffman é essencial para diversas aplicações modernas de compactação e transmissão de dados, como JPEG, MP3 e ZIP.

Data de Entrega: 06/12/2025

Valor Total: 10 pontos