

UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL
FACOM - FACULDADE DE COMPUTAÇÃO

ESTRUTURAS DE DADOS (2020/2)
PROFA. BIANCA DE ALMEIDA DANTAS

Trabalho Prático – Compactação de Dados

1 Descrição

Nos dias de hoje, com a crescente necessidade de mais espaço para armazenamento de dados, a utilização de técnicas de compactação de dados são de fundamental importância para facilitar a manipulação de arquivos grandes.

O objetivo deste trabalho é implementar as funcionalidades de compactação e descompactação, utilizando a linguagem C++, do algoritmo de Huffman para compactação de dados.

1.1 Formato da Entrada e da Saída

O alfabeto utilizado para os testes e que deve ser utilizado nas implementações é composto por letras, dígitos e espaços em branco. Ao invocar o seu programa, deverá ser especificado o nome de um arquivo de entrada. Por exemplo, se o seu programa se chamar *compactador* e se o arquivo de entrada for *entrada.txt*, o programa será invocado com a seguinte linha de comando:

./compactador entrada.txt

O arquivo deverá ser compactado pelo algoritmo implementado e o resultado da compactação deve ser escrito em um arquivo de saída com extensão **.freq**. Após essa primeira fase, o arquivo gerado deve ser descompactado usando a codificação gerada e, então, o resultado deve ser escrito em um novo arquivo com extensão **.defreq**. Note que esse último arquivo **DEVE** ser igual ao de entrada.

Além dos arquivos de saída, o seu programa deve imprimir **na tela** a taxa de compressão alcançada pelo algoritmo, ou seja, a porcentagem de redução no tamanho do arquivo. Para isso considere como se os códigos de Huffman fossem compostos por dígitos binários, ou seja, cada símbolo do código ocupa apenas um bit, enquanto cada símbolo da entrada é um caractere (que ocupa 8 bits).

O seu programa será compilado utilizando o compilador **g++** do Linux.

2 Grupos e Entrega

O trabalho pode ser feito em grupos de no máximo 4 estudantes e a entrega deve ser realizada até o dia **04/12/2020** em um arquivo compactado através de postagem no tópico adequado da página do AVA da disciplina.

Devem ser entregues os códigos fonte implementados e um arquivo *readme* contendo os nomes dos integrantes do grupo.

3 Avaliação

- Programas que não compilarem ou não gerarem os arquivos de resultado não serão considerados e receberão nota zero.
- A correção será realizada utilizando um conjunto de arquivos. Cada arquivo de entrada será compactado pelo algoritmo e, então, o arquivo compactado gerado será descompactado e comparado ao original. Será considerado totalmente correto o teste cujo resultado do processo de compactação/descompactação for igual à entrada.
- Somente a igualdade entre os arquivos fará com que o algoritmo receba a pontuação de cada teste.
- **Trabalhos copiados ou fortemente inspirados em outros receberão nota zero.**