

Redes de Computadores I

Trabalho prático 3 – Compactador LZW

1. Autores

- Daniele Harumi Ito – GRR20101243
- Paulo Ricardo Gemniczak – GRR20096237

2. Compilação e execução

- Compilação:
make
- Compactação:
java Programa -c entrada.txt saida.txt
- Descompactação:
java Programa -d saida.txt resultado.txt

3. Projeto

Arquivos:

- TipoDicionario.java
Classe contém dois atributos, “elemento” que representa um elemento do dicionário e “índice” que representa seu índice.
- Dicionario.java
Classe contém dois atributos. Um array de “TipoDicionario” e um int que guarda o tamanho do array conforme os elementos são inseridos. Também contém um método para inserir um registro, um para buscar o índice de um elemento, e outro para buscar o elemento de um índice específico no dicionário.
- ManipulacaoArquivo.java
Classe contém métodos que fazem leitura e escrita em arquivos. O método “lerArquivoChar” é utilizado para fazer a leitura do arquivo que será compactado. “salvarArquivoBytes” salva o arquivo compactado com inteiros de 16 bits. “lerArquivoBytes” lê o arquivo compactado de dois em dois bytes para gerar o array de inteiros. “salvarArquivoChar” salva o arquivo descompactado. Também possui métodos que convertem um array de bytes para inteiro e um inteiro para array de bytes.
- Lzw.java
Classe contém um atributo do tipo “Dicionario” e outro do tipo “ManipulacaoArquivo”. Também possui os métodos “compactar” e “descompactar” onde foram implementados os algoritmos vistos em sala de aula para compactação e descompactação lzw.
- Programa.java
Classe que possui o método “main” do projeto, que apenas verifica os argumentos passado na execução do programa, e chama o método correspondente ao que o usuário solicitou.