

Relatório - INF1018 - Software Básico (2023.2)

Primeiro Trabalho - Conversão entre codificações UTF-8 e varint

Nome do Aluno: Thiago Pereira Camerato

Matrícula: 2212580

Introdução

Este relatório descreve a implementação do primeiro trabalho da disciplina INF1018 - Software Básico. O objetivo deste trabalho é criar duas funções em linguagem C, `utf2varint` e `varint2utf`, que realizam a conversão de arquivos entre as codificações UTF-8 e varint.

Implementação

`utf2varint`

A função `utf2varint` foi implementada com sucesso. Ela lê um arquivo de entrada codificado em UTF-8, extrai os valores dos codepoints representados e grava esses valores em um arquivo de saída codificados como varints. A função realiza a leitura e gravação de arquivos binários e trata erros de E/S corretamente.

`varint2utf`

A função `varint2utf` também foi implementada com sucesso. Ela lê um arquivo de entrada contendo valores inteiros codificados como varints, obtém os valores inteiros originais e grava esses valores em um arquivo de saída codificados em UTF-8. Assim como a função `utf2varint`, a função `varint2utf` realiza a leitura e gravação de arquivos binários e trata erros de E/S adequadamente.

Testes

Utilizamos uma abordagem de Test Driven Development (TDD) para implementar as funções. Escrevemos testes antes de criar o código. `utf2varint`

Foram implementados casos de teste para a função `utf2varint` para garantir que ela converte corretamente arquivos codificados em UTF-8 para o formato varint. Aqui estão alguns exemplos de casos de teste utilizados:

Caso de Teste 1: Arquivo UTF-8 Vazio

Descrição: Um arquivo UTF-8 vazio é fornecido como entrada.

Resultado Esperado: A função deve gerar um arquivo varint vazio como saída.

Caso de Teste 2: Arquivo UTF-8 com Caracteres ASCII

Descrição: Um arquivo UTF-8 contendo apenas caracteres ASCII é fornecido como entrada.

Resultado Esperado: A função deve gerar um arquivo varint correspondente aos valores dos caracteres ASCII.

Caso de Teste 3: Arquivo UTF-8 com Caracteres Multibyte

Descrição: Um arquivo UTF-8 com caracteres multibyte (caracteres não ASCII) é fornecido como entrada.

Resultado Esperado: A função deve converter com precisão os caracteres multibyte para varints e criar um arquivo de saída correspondente.

Caso de Teste 4: Erro de Leitura de Arquivo

Descrição: A função recebe um arquivo de entrada que não pode ser lido (por exemplo, inexistente).

Resultado Esperado: A função deve tratar o erro de leitura do arquivo e retornar -1.

Caso de Teste 5: Erro de Escrita de Arquivo

Descrição: A função é incapaz de escrever no arquivo de saída.

Resultado Esperado: A função deve tratar o erro de escrita do arquivo e retornar -1.

varint2utf

A função varint2utf foi testada para garantir que ela converta adequadamente arquivos varint de volta para a codificação UTF-8. Abaixo estão alguns exemplos de casos de teste utilizados:

Caso de Teste 1: Arquivo varint Vazio

Descrição: Um arquivo varint vazio é fornecido como entrada.

Resultado Esperado: A função deve gerar um arquivo UTF-8 vazio como saída.

Caso de Teste 2: Arquivo varint com Valores ASCII

Descrição: Um arquivo varint contendo valores de caracteres ASCII é fornecido como entrada.

Resultado Esperado: A função deve converter com precisão os valores ASCII para caracteres UTF-8 correspondentes.

Caso de Teste 3: Arquivo varint com Valores Multibyte

Descrição: Um arquivo varint contendo valores de caracteres multibyte é fornecido como entrada.

Resultado Esperado: A função deve converter com precisão os valores multibyte em caracteres UTF-8 e criar um arquivo de saída correspondente.

Caso de Teste 4: Erro de Leitura de Arquivo

Descrição: A função recebe um arquivo de entrada que não pode ser lido (por exemplo, inexistente).

Resultado Esperado: A função deve tratar o erro de leitura do arquivo e retornar -1.

Caso de Teste 5: Erro de Escrita de Arquivo

Descrição: A função é incapaz de escrever no arquivo de saída.

Resultado Esperado: A função deve tratar o erro de escrita do arquivo e retornar -1.

Os casos de teste foram implementados para garantir que as funções utf2varint e varint2utf funcionem corretamente e tratem adequadamente erros de E/S.

Resultados

Ambas as funções `utf2varint` e `varint2utf` passaram em nossos testes e funcionam corretamente. Elas são capazes de converter arquivos entre as codificações UTF-8 e varint de forma eficiente e sem erros.

Conclusão

Este trabalho nos permitiu compreender melhor as codificações UTF-8 e varint, bem como a manipulação de arquivos em linguagem C. A abordagem de TDD também demonstrou ser uma prática eficaz para o desenvolvimento de software.

Considerações Finais

O trabalho foi realizado de acordo com as especificações fornecidas, e ambas as funções `utf2varint` e `varint2utf` estão funcionando conforme o esperado. As entregas foram feitas de acordo com as orientações do Moodle.