Faculdade de Design, Tecnologia e Comunicação
Universidade Europeia

Linux Operating System Utilities

Elementos do Grupo:

Carla Ribeiro - 20220411

Julia Disconzi - 20220007

Tiago Costa – 20190887

Repositório GitHub: https://github.com/tiagodccosta/OS-Compilers-PBL

Descrição do Problema

Os sistemas operativos baseados em Linux utilizam um conjunto de utilitários essenciais para a gestão de arquivos, processos e comunicação entre sistemas. Estes utilitários, muitos deles parte do GNU Coreutils, são fundamentais para administração e operação do sistema. Este projeto tem como objetivo desenvolver uma versão alternativa de alguns destes utilitários essenciais, explorando a sua implementação na linguagem C.

Casos de Uso

Administradores de Sistemas - Necessitam de comandos eficientes para gestão de ficheiros e processos.
Desenvolvedores - Utilizam comandos do terminal para manipulação de arquivos e gestão de versões.
Estudantes de Sistemas Operativos - Podem usar este projeto para compreender o funcionamento interno de utilitários do Linux.

Descrição da Solução

Descrição Genérica

Este projeto visa desenvolver e implementar seis utilitários do Linux em C, replicando suas funcionalidades conforme encontrado no GNU Coreutils. Os comandos selecionados para implementação são:

- echo : exibe o diretório no terminal
- pwd: mostra o diretório atual
- ls: lista arquivos e diretórios
- mkdir: cria um diretório
- rmdir: remove diretórios vazios
- cat - exibe o conteúdo de um arquivo

Enquadramento nas Áreas da Unidade Curricular

Sistemas Operativos: Desenvolvimento de aplicações interagindo com o kernel.
Programação em C: Manipulação de ficheiros, processos e gestão de memória.
Desenvolvimento de Software de Baixo Nível: Implementação eficiente de comandos de sistema.

Requisitos Técnicos

- Implementação em C.
- Compatibilidade com distribuições Linux (Ubuntu, Debian, Arch, etc.).
- Uso da API POSIX para manipulação de ficheiros e processos.
- Testes de performance e compatibilidade.

Arquitetura da Solução

Cada utilitário será implementado como um programa independente, compilado separadamente. O projeto seguirá uma estrutura modular, separando funções comuns para reutilização entre os utilitários.

Tecnologias a Utilizar

- **Linguagem:** C
- **Compilador:** GCC (GNU Compiler Collection)
- **Ferramentas de Desenvolvimento:** Makefile, GDB (para debugging), Valgrind (para análise de memória)
- **Controle de Versão:** GitHub

Planeamento e Calendarização

Distribuição de Tarefas

Semana 1-3 - Análise e Planeamento

a. Definição dos requisitos e escolha dos comandos a implementar
b. Criar estrutura do repositório
c. Definir arquitetura e módulos comuns

Semana 4-7 - Desenvolvimento Inicial

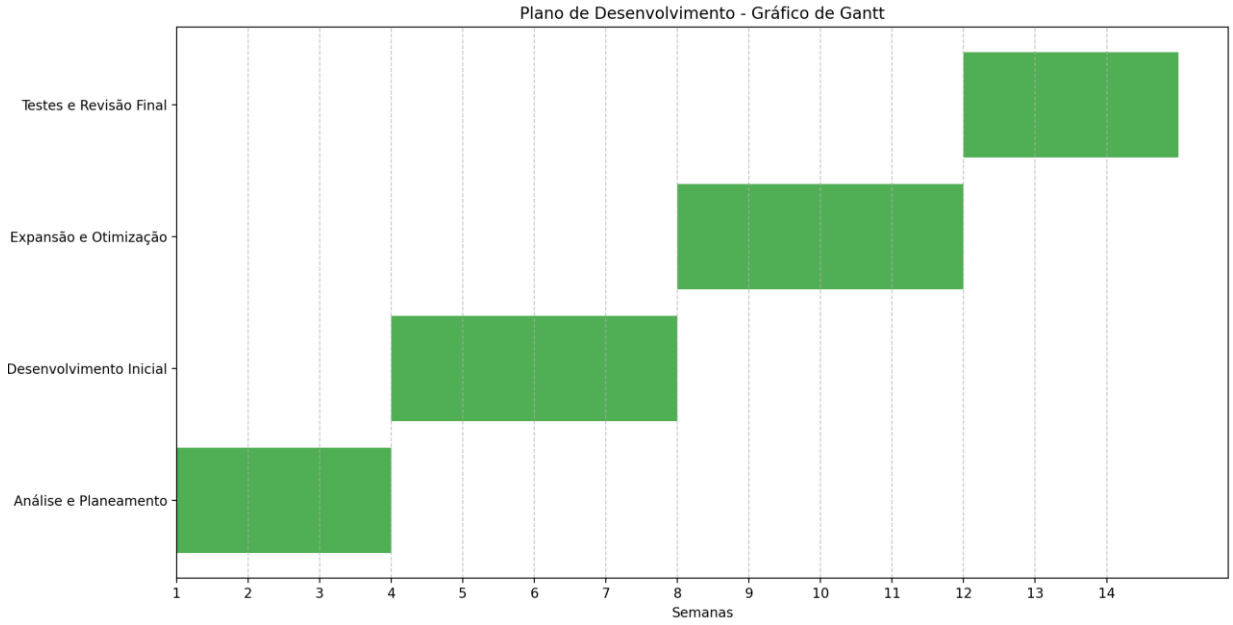
d. Implementação dos primeiros comandos
e. Testes unitários

Semana 8-11 - Expansão e Otimização

f. Melhorias de performance
g. Implementação de mais comandos
h. Documentação do código

Semana 12+ - Testes e Revisão Final

i. Testes completos de compatibilidade
j. Validação de performance
k. Entrega final do projeto



Bibliografia

- GNU Coreutils: https://www.gnu.org/software/coreutils/
- The Linux Programming Interface - Michael Kerrisk
- Advanced Programming in the UNIX Environment - W. Richard Stevens
- POSIX Programmer's Guide - Donald Lewine