



Universidade Federal de São João del Rei

Departamento de Ciência da Computação

Sistemas Operacionais

Professor: Rafael Sachetto

Relatório Trabalho Prático 1

Relatório do primeiro Trabalho Prático para a disciplina de Sistemas Operacionais do Bacharelado em Ciência da Computação da Universidade Federal de São João del Rei.

Filipe Mateus

Gustavo Detomi

Marcos Martins

Maio
2022

Conteúdo

1	Introdução	1
2	Resumo do projeto	2
3	Decisões de projeto	4
4	Bugs conhecidos ou problemas	5
	Anexo	6

1 Introdução

Para este Trabalho Prático foi implementado um interpretador de comandos que foi chamado ***shellso***, e conta com diversas funcionalidades comuns aos interpretadores de comandos: disparar processos, encadear a comunicação entre eles usando pipes, executar processos em segundo plano, listar processos disparados pelo interpretador, retornar processos do segundo plano de volta para o interpretador, entre outras.

2 Resumo do projeto

Algumas estruturas importantes:

```
struct shell {
    bool running;
    bool verbose;
    prompt_function prompt;
    aliases* aliases;
    background_jobs* jobs;
    shell_builtin_commands* builtin_commands;
}
```

A estrutura *shell* representa a base do nosso interpretador de comandos, e deve manter registro dos jobs, dos comandos próprios e dos *aliases*.

```
struct command {
    int argc;
    char** argv;
    command* next;
    command_chain_type chain_type;
    char* stdin_file_redirection;
    char* stdout_file_redirection;
    char* stderr_file_redirection;
}
```

A estrutura *shell* representa os comandos em si, possui os campos necessários para armazenar informações do comando, do encadeamento de comandos e redirecionamento de entrada e saída.

```
struct builtin_command {
    char* command;
    builtin_command_function function;
}
```

A estrutura *builtin_command* é usada para representar os comandos implementados internamente no *shellso*.

```
struct _shell_builtin_commands {
    builtin_command** commands;
    int size;
    int count;
} shell_builtin_commands;
```

É uma tabela *Hash* contendo todos os comandos que foram implementados diretamente no *shell*.

```
struct background_job {  
    char* command;  
    int pid;  
    int status;  
    background_job* next;  
}
```

Essa estrutura representa os comandos que estão sendo executados em segundo plano pelo ***shellso***.

3 Decisões de projeto

O *shellso* foi projetado de forma a ser dividido em módulos bem separados com o objetivo de tornar seu desenvolvimento e eventual manutenção mais fácil. Cada elemento do interpretador foi implementado em arquivos separados dentro da pasta *shell*, algumas funções utilitárias para trabalhar com cadeias de caracteres foram implementadas na pasta *string*.

Essa decisão de dividir o programa em módulos primeiramente divide bem as responsabilidades, o que garante que a pessoa desenvolvedora possa trabalhar em um pedaço pequeno do problema simplificando o projeto e a implementação, além disso auxilia a evitar efeitos colaterais, já que se ao longo de todo o problema as estruturas apresentadas na seção anterior fossem visíveis e alteráveis seria bastante provável que ao longo do desenvolvimento e eventuais manutenções fossem provocados efeitos colaterais ou seja, alterações introduzidas sem cuidado causando modificações incompletas nas estruturas de dados, provocando erros difíceis de reproduzir e detectar.

Outra decisão que tomamos foi de implementar testes, para isso usamos a biblioteca Criterion, disponível em <https://criterion.readthedocs.io/en/master/intro.html>. Com essa simples biblioteca de testes foi possível escrever testes de unidades que facilitam o desenvolvimento tornando mais fácil visualizar quando um código novo resolve o problema a que se propõe sem causar efeitos colaterais em outras partes do código.

4 Bugs conhecidos ou problemas

Anexo

Repositório: <https://github.com/marcosfons/shellso> Documentação on-line: <https://marcosfons.github.io/shellso/>