

Alunos: Chrystian de Sousa Guth
Lucas Pereira da Silva
Disciplina: Sistemas Operacionais I – INE5412
Professor Luis Fernando Friedrich

Matrículas: 10103131
10100754

Roteiro de Laboratório 3

Exercício 3.1 Micro-shell

Descrição

Em **lab3-1.c** foi implementado um micro-shell capaz de reconhecer e executar comandos padrão do Unix com suas opções e seus argumentos. O comando “exit” finaliza a execução do programa.

Para compilá-lo utilize “**gcc -o lab31 lab3-1.c**” e para executá-lo utilize “**./lab31**”.

Código em C

```
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>
#include <stdio.h>
#include <string.h>

char input[1024];
char pasta[4096];
char *argv[10];

int main(void) {
    int i, j;
    char *pch;
    while(1) {
        for(i = 0; i < 10; i++) // zera o vetor de argumentos
            argv[i] = NULL;

        printf("mshell> ");
        fgets(input, 1024, stdin);
        input[strlen(input)-1] = '\0';

        if(!strcmp(input, "exit")) // vê se é exit
            exit(0);

        j = 0;
        pch = strtok(input, " ");

        while(pch != NULL && j < 10) { // pega até 10 parâmetros
            argv[j] = pch;
            pch = strtok(NULL, " ");
            j++;
        }

        if(!strcmp(argv[0], "cd")) { // checa se é cd
            if(chdir(argv[1]) != 0)
                printf("caminho inválido\n");
        }
        else {
            if(fork() == 0) {
```

```

        if(!strcmp(argv[0], "ls")) // deixa o ls colorido sempre
            argv[1] = "--color=auto";

        if(execvp(argv[0], argv) == -1){ // executa o comando
            printf("comando inválido\n");
            exit(1);
        }
    }else
        wait(NULL);
    }
}
}

```

Exercício 3.2 Temporizador usando sinais

Descrição

Em **lab3-2.c**, foi implementado um programa que espera pela entrada do usuário e termina se depois de um determinado tempo (segundos) nenhuma entrada for recebida.

Para compilá-lo utilize “**gcc -o lab32 lab3-2.c**” e para executá-lo utilize “**./lab32 x**”, onde x é o tempo de espera (segundos) do programa em relação a uma entrada do usuário. Se x for vazio → x=0, então o programa encerra-rá logo depois de iniciar.

Código em C

```

#include <signal.h>
#include <stdio.h>
#include <unistd.h>

void timeout_function()
{
    printf("Tempo esgotado (timeout)!\n");
    exit(1);
}

int main(int argc, char* argv[]){
    char a[100];
    int tempo;

    if(argv[1]==NULL)
        timeout_function();

    tempo = atoi(argv[1]);

    if(tempo == 0)
        timeout_function();

    signal(SIGALRM, timeout_function);

    while(1) {

```

```
        alarm(tempo);
        printf("Digite uma entrada: \n");
        scanf("%s", &a);
        alarm(0);
    }

    return 0;
}
```