Московский авиационный институт
(Национальный исследовательский университет)
Факультет "Информационные технологии и прикладная математика"
Кафедра "Вычислительная математика и программирование"

Лабораторная работа №1 по курсу “Операционные системы”

Студент: Слетюрин Кирилл Сергеевич

Группа: М8О-208Б-22

Преподаватель: Миронов Евгений Сергеевич

Вариант: 5

Оценка: _____

Дата: _____

Подпись: _____

Москва, 2023

Содержание

1 Репозиторий

https://github.com/kirill483/OS

2 Цель работы

Приобретение практических навыков в:

- Управлении процессами в ОС
- Обеспечении обмена данных между процессами посредством каналов

3 Задание

Составить и отладить программу на языке Си, осуществляющую работу с процессами и взаимодействие между ними в одной из двух операционных систем. В результате работы программа (основной процесс) должен создать для решение задачи один или несколько дочерних процессов. Взаимодействие между процессами осуществляется через системные сигналы/события и/или каналы (pipe). Необходимо обрабатывать системные ошибки, которые могут возникнуть в результате работы.

4 Описание работы программы

Родительский процесс создает дочерний процесс. Первой строчкой пользователь в консоль родительского процесса пишет имя файла, которое будет передано при создании дочернего процесса. Родительский и дочерний процесс должны быть представлены разными программами. Родительский процесс передает команды пользователя через pipe1, который связан с стандартным входным потоком дочернего процесса. Дочерний процесс принеобходимости передает данные в родительский процесс через pipe2. Результаты своей работы дочерний процесс пишет в созданный им файл. Допускается просто открыть файл и писать туда, не перенаправляя стандартный поток вывода.

В ходе выполнения лабораторной работы я использовала следующие системные вызовы:

- fork() - создание нового процесса
- pipe() - создание канала
- dup2() - создание копии файлового дескриптора, используя для нового дескриптора самый маленький свободный номер файлового дескриптора.
- execlp() - запуск файла на исполнение

5 Исходный код

lab1.cpp

```
    #include <utils.hpp>

int ParentRoutine(const char* pathToChild, FILE* stream){
    int pipe1[2];
    int pipe2[2];
    create_pipe(pipe1);
    create_pipe(pipe2);
    char* s = NULL;
    size_t size = 0;
    int k;
    k = getline(&s, &size, stream);
    s[k - 1] = '\0';
    pid_t pid = create_process();
    if(pid == 0){
        close(pipe1[1]);
        close(pipe2[0]);
        dup2(pipe1[0], STDIN_FILENO);
        dup2(pipe2[1], STDOUT_FILENO);
        execl(pathToChild, s, NULL);
    }
    else{
        close(pipe1[0]);
        int n;
        while(1){
            read(pipe2[0], &n, sizeof(n));
            if(n == -1){
                close(pipe1[1]);
                close(pipe2[0]);
                close(pipe2[1]);
                return 0;
            }
            fscanf(stream, "%d", &n);
            write(pipe1[1], &n, sizeof(n));
        }
    }
}
```

utils.c

```
#include "utils.h"

#include <stdio.h>
#include <stdlib.h>

void ReverseString(char* string, size_t length) {
    for (size_t i = 0; i < length >> 1; ++i) {
        char temp = string[i];
        string[i] = string[length - i - 1];
        string[length - i - 1] = temp;
    }
}
```

child.cpp

```
#include <utils.hpp>

int main(int argc, char* argv[]){
```

```
    FILE* file = fopen(argv[0],"w");
    int n;
    int fl = 1;
    write(STDOUT_FILENO, &fl, sizeof(fl));
    while(1){
        read(STDIN_FILENO, &n, sizeof(n));
        if(check(n) == -1){
            fl = -1;
            write(STDOUT_FILENO, &fl, sizeof(fl));
            fclose(file);
            close(STDIN_FILENO);
            close(STDOUT_FILENO);
            return 0;
        }
        write(STDOUT_FILENO, &fl, sizeof(fl));
        fprintf(file,"%d\n", n);
    }
}
```

main.c

```
#include "lab1.hpp"

int main(void) {
    const char* st = "/mnt/c/Users/Kirill/OS/lab1/build/child";
    ParentRoutine(st, stdin);
}
```

6 Тесты

```
#include <gtest/gtest.h>
#include <utils.hpp>
#include <lab1.hpp>

bool is_correct(int* input,int in_size, int* output, int out_size)
    {
     FILE* file1 = fopen("input.txt","w");
     fprintf(file1, "%s\n", "output.txt");
     for(int i = 0; i < in_size; i++){
         fprintf(file1, "%d\n", input[i]);
     }
     fclose(file1);
     file1 = fopen("input.txt","r");
     ParentRoutine("/mnt/c/Users/Kirill/OS/lab1/build/child", file1
    );
     FILE* file2 = fopen("output.txt","r");
     int tmp;
     for(int i = 0; i < out_size; i++){
         fscanf(file2, "%d\n", &tmp);
         if(tmp != output[i]){
             fclose(file2);
             return false;
         }
     }
     fclose(file2);
     return true;
}


TEST(input5_output4, test1){
     int in_size = 5;
     int out_size = 4;
     int input[5] = {4,4,6,9,-1};
     int output[4] = {4,4,6,9};
     ASSERT_TRUE(is_correct(input, in_size, output, out_size));
}

TEST(input2_output1, test2){
     int in_size = 2;
     int out_size = 1;
     int input[2] = {110,-15};
     int output[1] = {110};
     ASSERT_TRUE(is_correct(input, in_size, output, out_size));
}

TEST(input5_output0, test3){
     int in_size = 5;
     int out_size = 0;
     int input[5] = {-12,0,6,9,-1};
     int output[0] = {};
     ASSERT_TRUE(is_correct(input, in_size, output, out_size));
}

TEST(input3_output2, test4){
     int in_size = 3;
     int out_size = 2;
     int input[5] = {18,100,1};
```

```
56      int output[4] = {18,100};
57      ASSERT_TRUE(is_correct(input, in_size, output, out_size));
58  }
59
60  TEST(input4_output4, test5){
61      int in_size = 4;
62      int out_size = 4;
63      int input[5] = {4,4,6,150};
64      int output[4] = {4,4,6,150};
65      ASSERT_TRUE(is_correct(input, in_size, output, out_size));
66  }
67
68  int main(int argc, char **argv) {
69      testing::InitGoogleTest(&argc, argv);
70      return RUN_ALL_TESTS();
71  }
```

7 Демонстрация работы программы

```
kirill@DESKTOP-7E05ERB:/mnt/c/Users/Kirill/OS/lab1/build$ ./main
file.txt
4
6
-1
kirill@DESKTOP-7E05ERB:/mnt/c/Users/Kirill/OS/lab1/build$ cat file.txt
4
6
```

8 Запуск тестов

```
kirill@DESKTOP-7E05ERB:/mnt/c/Users/Kirill/OS/lab1/build$ ./Test
[==========] Running 5 tests from 5 test suites.
[----------] Global test environment set-up.
[----------] 1 test from input5_output4
[ RUN      ] input5_output4.test1
[       OK ] input5_output4.test1 (13 ms)
[----------] 1 test from input5_output4 (13 ms total)

[----------] 1 test from input2_output1
[ RUN      ] input2_output1.test2
[       OK ] input2_output1.test2 (6 ms)
[----------] 1 test from input2_output1 (6 ms total)

[----------] 1 test from input5_output0
[ RUN      ] input5_output0.test3
[       OK ] input5_output0.test3 (9 ms)
[----------] 1 test from input5_output0 (9 ms total)

[----------] 1 test from input3_output2
[ RUN      ] input3_output2.test4
[       OK ] input3_output2.test4 (11 ms)
[----------] 1 test from input3_output2 (11 ms total)

[----------] 1 test from input4_output4
[ RUN      ] input4_output4.test5
[       OK ] input4_output4.test5 (9 ms)
[----------] 1 test from input4_output4 (9 ms total)

[----------] Global test environment tear-down
[==========] 5 tests from 5 test suites ran. (48 ms total)
[  PASSED  ] 5 tests.
```

9 Выводы

В результате выполнения данной лабораторной работы была написана программа на языке C++, осуществляющая работу с процессами и взаимодействие между ними. Преобретены практические навыки в управлении процессами в ОС и обеспечении обмена данных между процессами посредством каналов.