Московский Авиационный Институт

(Национальный Исследовательский Университет)

Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

**Лабораторная работа №1 по курсу**

**«Операционные системы»**

**ВЗАИМОДЕЙСТВИЕ МЕЖДУ ПРОЦЕССАМИ**

Студент: Муратов Артём Алексеевич

Группа: М8О–212Б–20

Вариант: 9

Преподаватель: Соколов Андрей Алексеевич

Оценка: \_\_\_\_\_\_\_\_\_\_

Дата: \_\_\_\_\_\_\_\_\_\_

Подпись: \_\_\_\_\_\_\_\_\_\_

Москва, 2023.

**Постановка задачи**

## Цель работы

Целью является приобретение практических навыков в:

* Управление процессами в ОС
* Обеспечение обмена данными между процессами посредством труб

## 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.

Родительский процесс создает дочерний процесс. Первой строчкой пользователь в консоль родительского процесса вводит имя файла, которое будет использовано для открытия файла с таким именем на чтение. Стандартный поток ввода дочернего процесса переопределяется открытым файлом. Дочерний процесс читает команды из стандартного потока ввода. Стандартный поток вывода дочернего процесса перенаправляется в pipe1. Родительский процесс читает из pipe1 и прочитанное выводит в свой стандартный поток вывода. Родительский и дочерний процесс должны быть представлены разными программами.

9 вариант) В файле записаны команды вида:«число число число <endline>». Дочерний процесс производит деление первого числа команда, на последующие числа в команде, а результат выводит в стандартный поток вывода. Если происходит деление на 0, то тогда дочерний и родительский процесс завершают свою работу. Проверка деления на 0 должна осуществляться на стороне дочернего процесса. Числа имеют тип float. Количество чисел может быть произвольным.

**Общие сведения о программе**

Программа компилируется из файла main.c. Для сборки используется CMake. Также используется заголовочные файлы:

#include <string.h>

#include <unistd.h>

#include <stdio.h>

#include <stdlib.h>

#include <sys/types.h>

#include <fcntl.h>

#include <sys/wait.h> .

В программе используются следующие системные вызовы:

1. **read** — считывает с консоли
2. **write** — записывает в консоль
3. **fork** — создаёт дочерний процесс, который повторяет программу родительского процесса.
4. **pipe** — создаёт одностороннюю трубу для передачи данных между процессами.
5. **exit** — завершает программу
6. **open** — открывает файл.
7. **close** — закрывает файл
8. **dup2** — переназначение файлового дескриптора.
9. **execvp** — запускает исполняемый файл изнутри другой программы.

**Общий метод и алгоритм решения**.

Для реализации поставленной задачи необходимо:

1. Изучить принципы работы dlsym, dlopen, dlclose.
2. Написать библиотеку vecmd5, для работы с вектором из md5 сумм.
3. Организовать простейший командный интерфейс в файлах test1.c и test2.c.
4. В файле test1.c подключить библиотеку на этапе компиляции.
5. В файле test2.c загрузить библиотечные функции в runtime, с помощью dlsym, dlopen, dlclose.

**Основные файлы программы**

[Исходники; не рекомендуется использовать большой междустрочный интервал и подсветку синтаксиса]

**Main.c**

.....

**Сhild.c**

…...

**CMakeLists.txt**

……

**Пример работы**

**1.** 10 2 5

virtbox@virtbox-VirtualBox:~/OS\_labs/lab\_os\_1/build$ ./main

/home/virtbox/OS\_labs/lab\_os\_1/tests/test\_1.txt

1

**2.** 120 2 5 4 3

virtbox@virtbox-VirtualBox:~/OS\_labs/lab\_os\_1/build$ ./main

/home/virtbox/OS\_labs/lab\_os\_1/tests/test\_2.txt

1

**3.** 2 0

virtbox@virtbox-VirtualBox:~/OS\_labs/lab\_os\_1/build$ ./main

/home/virtbox/OS\_labs/lab\_os\_1/tests/test\_3.txt

Division by 0

**4.** 127 34 23 3 5

virtbox@virtbox-VirtualBox:~/OS\_labs/lab\_os\_1/build$ ./main

/home/virtbox/OS\_labs/lab\_os\_1/tests/test\_4.txt

0.0108269397169

**5.** 0

virtbox@virtbox-VirtualBox:~/OS\_labs/lab\_os\_1/build$ ./main

/home/virtbox/OS\_labs/lab\_os\_1/tests/test\_5.txt

0

**6.** 0 1234 5678 90

virtbox@virtbox-VirtualBox:~/OS\_labs/lab\_os\_1/build$ ./main

/home/virtbox/OS\_labs/lab\_os\_1/tests/test\_6.txt

0

**Вывод**

[Очень важный раздел. Вода здесь может негативно повлиять на оценку]