Московский Авиационный Институт

(Национальный Исследовательский Университет)

Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

**Лабораторная работа №1 по курсу**

**«Операционные системы»**

Студент: Казанцев Данила Игоревич

Группа: М8О-207Б-21

Преподаватель: Миронов Евгений Сергеевич

Оценка: \_\_\_\_\_\_\_\_\_\_\_

Дата: \_\_\_\_\_\_\_\_\_\_\_

Подпись: \_\_\_\_\_\_\_\_\_\_

Москва, 2022

**Репозиторий**

https://github.com/thgdanilaya/lab1

**Постановка задачи**

**Цель работы**

Научиться работать с strace.

**Общие сведения о программе**

Описание работы strace

execve — открывает исполняемый файл

brk — определение конца сегмента данных для процесса

arch\_prctl - задаёт состояние процесса или нити, зависящие от архитектуры

openat — открывает файл

fstat - считывает состояние файла

mmap, munmap - отражает файлы или устройства в памяти, снимает их отражение

mprotect - контролирует доступ к области памяти

read, write — чтение и запись

ПОДРОБНЕЕ:

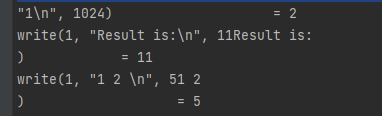
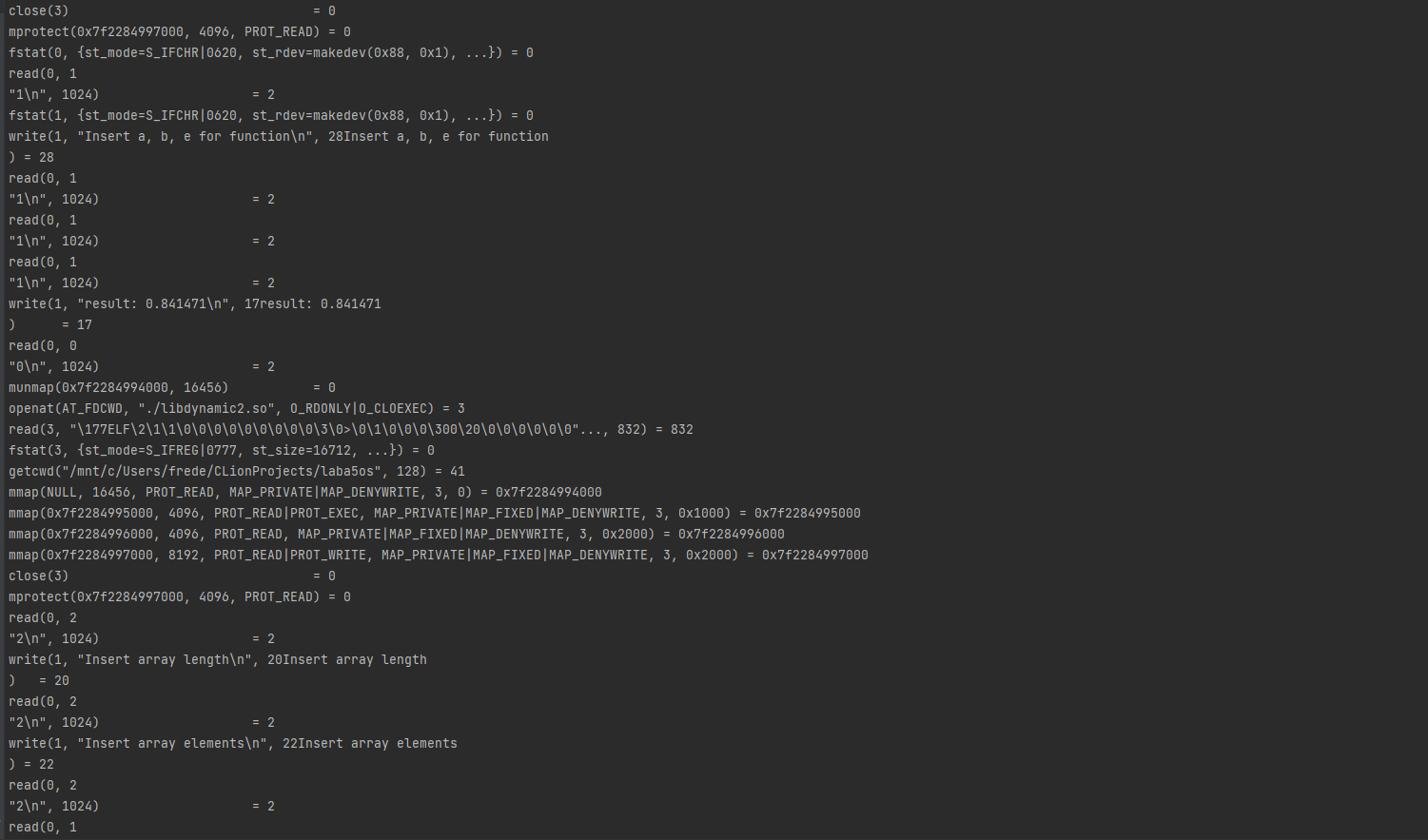
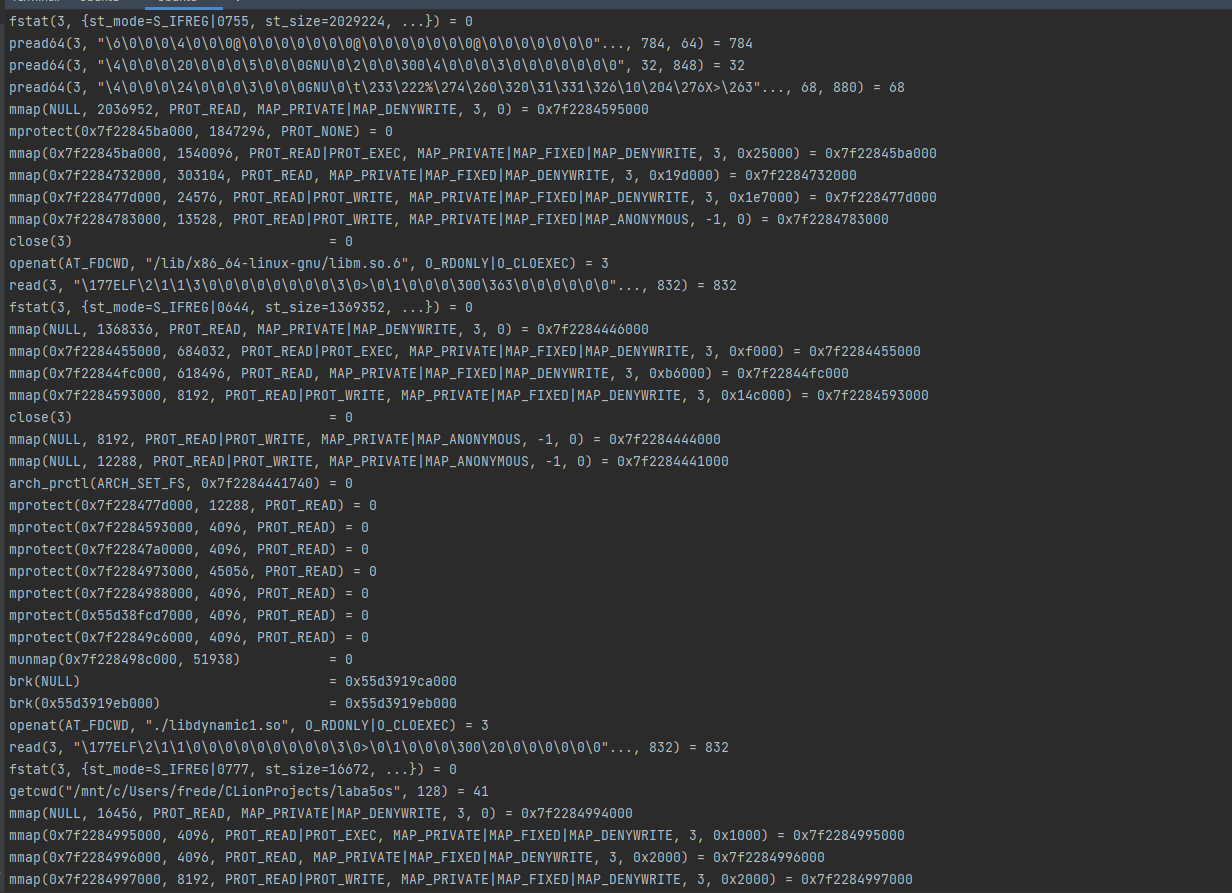
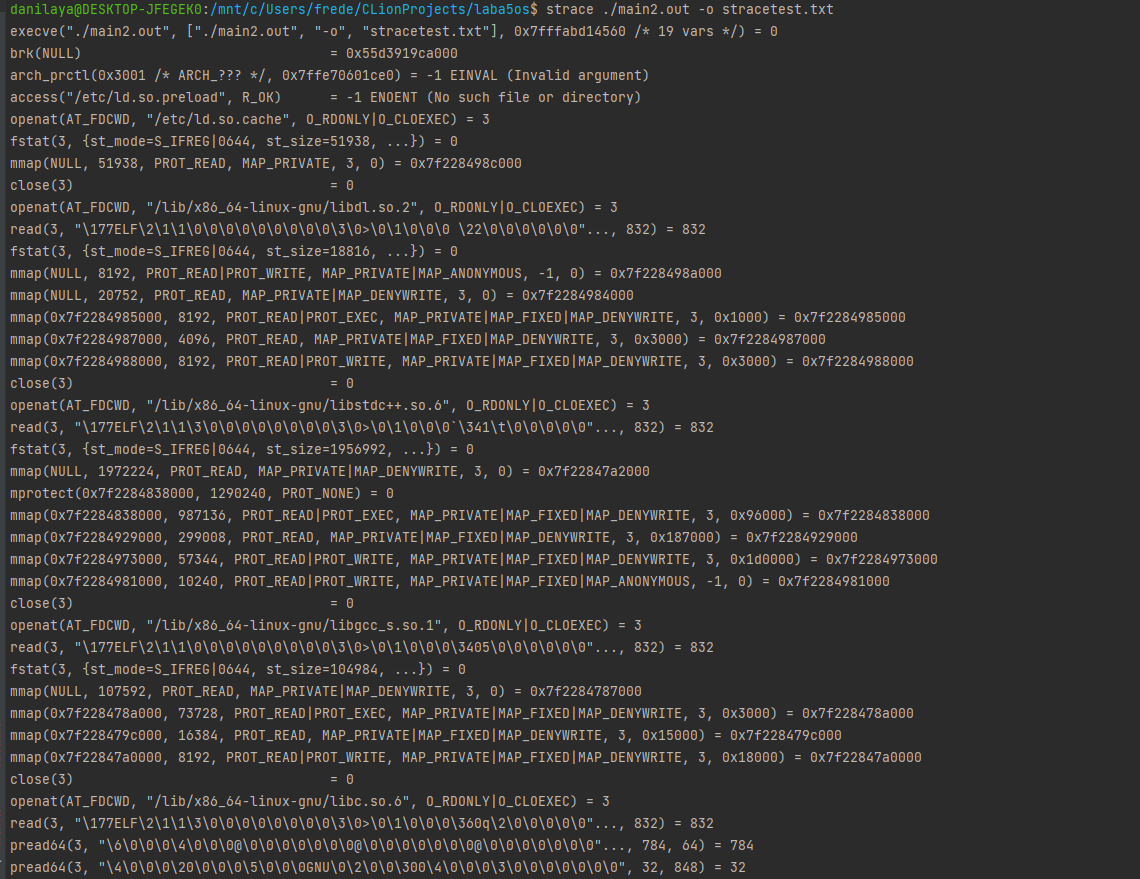
**void \*mmap(void \*addr, size\_t length, int prot, int flags, int fd, off\_t offset);** возвращает указатель на начало выделенного блока памяти. Addr — позволяет выбрать конкретный адрес, length — длина участвка, int prot — раззрешения (write, read), fd — файловый дескриптор, offset — сдвиг относительно адреса.

**int fstat(int filedes, struct stat \*buf);**

stat возвращает информацию о файле file\_name и заполняет буфер buf. fstat идентична stat, только возвращается информация об открытом файле, на который указывает filedes. Системные вызовы stat и fstat соответствуют SVr4, SVID, POSIX, X/OPEN, BSD 4.3. Системный вызов lstat соответствует 4.3BSD и SVr4. SVr4 документирует дополнительные коды ошибок fstat:EINTR,ENOLINKиEOVERFLOW.SVr4описываетдополнительныекодыошибок stat и lstat: EACCES, EINTR, EMULTIHOP, ENOLINK и EOVERFLOW.

**ssize\_t write(int fd, const void \*buf, size\_t count);**

write записывает до count байтов из буфера buf в файл, на который ссылается файловый описатель fd. POSIX указывает на то, что вызов write(), произошедший после вызова read() возвращает уже новое значение. Заметьте, что не все файловые системы соответствуют стандарту POSIX. В случае успешного завершения возвращается количество байтов, которые были записаны (ноль означает, что не было записано ни одного байта). В случае ошибки возвращается -1, а переменной errno присваивается соответствующее значение. Если count равен нулю, а файловый описатель ссылается на обычный файл, то будет возвращен ноль и больше не будет произведено никаких действий. Для специальных файлов результаты не могут быть перенесены на другую платформу.



**Выводы**

Я научился работать с strace и изучил системные вызовы.