Московский Авиационный Институт

(Национальный Исследовательский Университет)

Институт №8 “Компьютерные науки и прикладная математика”

Кафедра №806 “Вычислительная математика и программирование”

# Лабораторная работа №3 по курсу

# «Операционные системы»

Группа: М8О-210Б-23

Студент: Морозов Захар Олегович

Преподаватель: Бахарев В.Д. (ФИИТ)

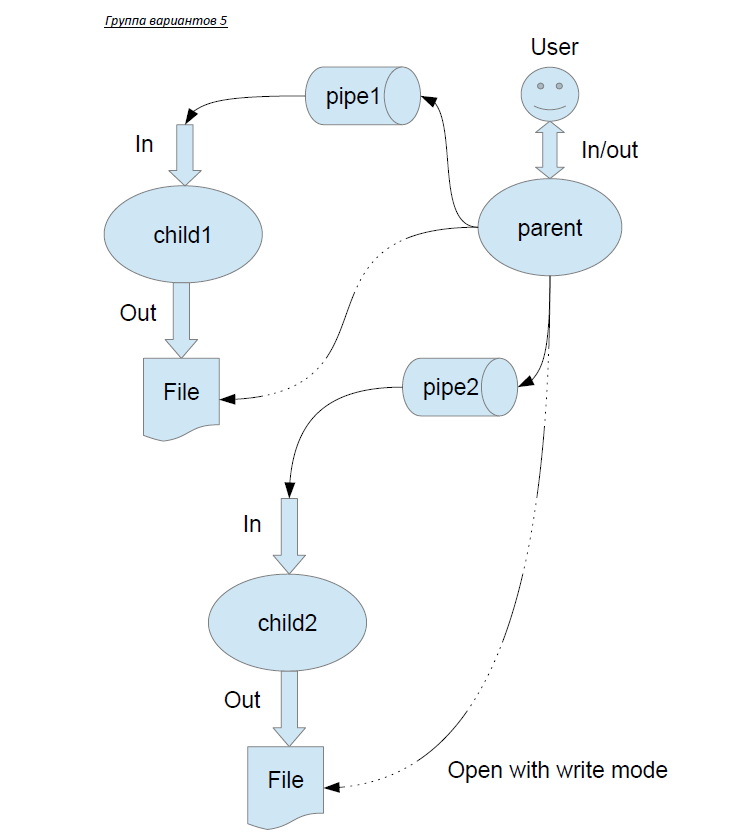
Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата: 06.01.25

Москва, 2025

# Постановка задачи

**Вариант 19.**



Пользователь вводит строки. Далее эти строки передаются от родительского процесса в дочерний. Дочерние процессы удаляют все гласные из строк.

# Общий метод и алгоритм решения

Использованные системные вызовы:

* mmap – отображение файла в память
* fork – создание дочернего процесса
* execl – замена исполняемого кода
* sem\_open – создание/подключение к семафору
* sem\_post – поднятие семафора
* sem\_wait – опускание семафора
* wait – ожидание завершения процесса
* sem\_unlink - уничтожает именованный семафор
* shm\_open – открывает объект разделяемой памяти
* ftruncate - укорачивает файл до указанной длины
* sem\_close – закрывает именованый семафор
* munmap – снимает отражение файла

# Код программы

**parent.c**

#include <unistd.h>

#include <string.h>

#include <stdio.h>

#include <stdlib.h>

#include <sys/types.h>

#include <sys/wait.h>

#include <fcntl.h>

#include <sys/mman.h>

#include <sys/stat.h>

#include <semaphore.h>

#define BUFFER\_SIZE 256

#define SHM\_SIZE 1024

int main() {

    char filename1[BUFFER\_SIZE];

    char filename2[BUFFER\_SIZE];

    ssize\_t bytes\_read;

    const char \*prompt1 = "Enter the file name for child1: ";

    write(STDOUT\_FILENO, prompt1, strlen(prompt1));

    bytes\_read = read(STDIN\_FILENO, filename1, BUFFER\_SIZE - 1);

    filename1[bytes\_read - 1] = '\0';

    const char \*prompt2 = "Enter the file name for child2: ";

    write(STDOUT\_FILENO, prompt2, strlen(prompt2));

    bytes\_read = read(STDIN\_FILENO, filename2, BUFFER\_SIZE - 1);

    filename2[bytes\_read - 1] = '\0';

    int file\_check1 = open(filename1, O\_WRONLY | O\_CREAT | O\_TRUNC, 0666);

    if (file\_check1 == -1) {

        const char \*error\_msg = "Error: could not open the file for child1\n";

        write(STDERR\_FILENO, error\_msg, strlen(error\_msg));

        perror("open");

        return 1;

    }

    close(file\_check1);

    int file\_check2 = open(filename2, O\_WRONLY | O\_CREAT | O\_TRUNC, 0666);

    if (file\_check2 == -1) {

        const char \*error\_msg = "Error: could not open the file for child2\n";

        write(STDERR\_FILENO, error\_msg, strlen(error\_msg));

        perror("open");

        return 1;

    }

    close(file\_check2);

    int shm\_fd1 = shm\_open("/shm1", O\_CREAT | O\_RDWR, 0666);

    int shm\_fd2 = shm\_open("/shm2", O\_CREAT | O\_RDWR, 0666);

    ftruncate(shm\_fd1, SHM\_SIZE);

    ftruncate(shm\_fd2, SHM\_SIZE);

    char \*shm\_ptr1 = mmap(0, SHM\_SIZE, PROT\_READ | PROT\_WRITE, MAP\_SHARED, shm\_fd1, 0);

    char \*shm\_ptr2 = mmap(0, SHM\_SIZE, PROT\_READ | PROT\_WRITE, MAP\_SHARED, shm\_fd2, 0);

    sem\_t \*sem1 = sem\_open("/sem1", O\_CREAT, 0666, 1);

    sem\_t \*sem2 = sem\_open("/sem2", O\_CREAT, 0666, 1);

    if (sem1 == SEM\_FAILED || sem2 == SEM\_FAILED) {

        perror("sem\_open");

        return 1;

    }

    pid\_t pid1 = fork();

    if (pid1 == -1) {

        const char \*error\_msg = "Error when creating a child process 1\n";

        write(STDERR\_FILENO, error\_msg, strlen(error\_msg));

        return 1;

    } else if (pid1 == 0) {

        execl("./child.out", "./child.out", filename1, "/shm1", "/sem1", NULL);

        const char \*error\_msg = "Error when starting child1\n";

        write(STDERR\_FILENO, error\_msg, strlen(error\_msg));

        exit(1);

    }

    pid\_t pid2 = fork();

    if (pid2 == -1) {

        const char \*error\_msg = "Error when creating a child process 2\n";

        write(STDERR\_FILENO, error\_msg, strlen(error\_msg));

        return 1;

    } else if (pid2 == 0) {

        execl("./child.out", "./child.out", filename2, "/shm2", "/sem2", NULL);

        const char \*error\_msg = "Error when starting child2\n";

        write(STDERR\_FILENO, error\_msg, strlen(error\_msg));

        exit(1);

    }

    char input[BUFFER\_SIZE];

    while (1) {

        bytes\_read = read(STDIN\_FILENO, input, sizeof(input) - 1);

        if (bytes\_read <= 0) {

            const char \*error\_msg = "Error while reading the string\n";

            write(STDERR\_FILENO, error\_msg, strlen(error\_msg));

            break;

        }

        if (bytes\_read > 0 && input[bytes\_read - 1] == '\n') {

            input[bytes\_read - 1] = '\0';

        }

        if (strcmp(input, "exit") == 0) {

            sem\_wait(sem1);

            strcpy(shm\_ptr1, input);

            sem\_post(sem1);

            sem\_wait(sem2);

            strcpy(shm\_ptr2, input);

            sem\_post(sem2);

            break;

        }

        int randomInt = rand() % 100;

        if (randomInt < 80) {

            sem\_wait(sem1);

            strcpy(shm\_ptr1, input);

            sem\_post(sem1);

        } else {

            sem\_wait(sem2);

            strcpy(shm\_ptr2, input);

            sem\_post(sem2);

        }

    }

    waitpid(pid1, NULL, 0);

    waitpid(pid2, NULL, 0);

    munmap(shm\_ptr1, SHM\_SIZE);

    munmap(shm\_ptr2, SHM\_SIZE);

    shm\_unlink("/shm1");

    shm\_unlink("/shm2");

    sem\_close(sem1);

    sem\_close(sem2);

    sem\_unlink("/sem1");

    sem\_unlink("/sem2");

    return 0;

}

**Child.c**

#include <unistd.h>

#include <fcntl.h>

#include <string.h>

#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

#include <sys/mman.h>

#include <semaphore.h>

#define BUFFER\_SIZE 256

#define SHM\_SIZE 1024

bool is\_vowel(char c) {

    c = (c >= 'A' && c <= 'Z') ? c + 32 : c;

    return c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u';

}

void remove\_vowels(char \*str) {

    int write\_index = 0;

    for (int read\_index = 0; str[read\_index] != '\0'; read\_index++) {

        if (!is\_vowel(str[read\_index])) {

            str[write\_index++] = str[read\_index];

        }

    }

    str[write\_index] = '\0';

}

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

    if (argc < 4) {

        const char \*error\_msg = "Error: The file name, shared memory, or semaphore was not passed\n";

        write(STDERR\_FILENO, error\_msg, strlen(error\_msg));

        return 1;

    }

    int file = open(argv[1], O\_WRONLY | O\_CREAT | O\_TRUNC, 0666);

    if (file == -1) {

        const char \*error\_msg = "Error: the file could not be opened for writing\n";

        write(STDERR\_FILENO, error\_msg, strlen(error\_msg));

        perror("open");

        return 1;

    }

    int shm\_fd = shm\_open(argv[2], O\_RDWR, 0666);

    if (shm\_fd == -1) {

        perror("shm\_open");

        return 1;

    }

    char \*shm\_ptr = mmap(0, SHM\_SIZE, PROT\_READ | PROT\_WRITE, MAP\_SHARED, shm\_fd, 0);

    if (shm\_ptr == MAP\_FAILED) {

        perror("mmap");

        return 1;

    }

    sem\_t \*sem = sem\_open(argv[3], 0);

    if (sem == SEM\_FAILED) {

        perror("sem\_open");

        return 1;

    }

    char buffer[BUFFER\_SIZE];

    while (1) {

        sem\_wait(sem);

        if (shm\_ptr[0] != '\0') {

            strcpy(buffer, shm\_ptr);

            if (strcmp(buffer, "exit") == 0) {

                sem\_post(sem);

                break;

            }

            shm\_ptr[0] = '\0';

            sem\_post(sem);

            remove\_vowels(buffer);

            if (write(file, buffer, strlen(buffer)) == -1 || write(file, "\n", 1) == -1) {

                const char \*error\_msg = "Error: failed to write to a file\n";

                write(STDERR\_FILENO, error\_msg, strlen(error\_msg));

                close(file);

                return 1;

            }

        } else {

            sem\_post(sem);

            usleep(100000);

        }

    }

    close(file);

    munmap(shm\_ptr, SHM\_SIZE);

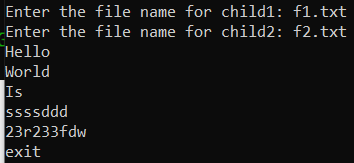
    sem\_close(sem);

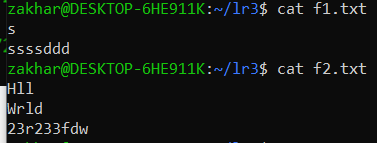
    return 0;

}

# Протокол работы программы

**Тестирование:**

****



**Strace:**

execve("./parent.out", ["./parent.out"], 0x7fff1f362218 /\* 22 vars \*/) = 0

brk(NULL) = 0x56318445c000

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7f841c409000

access("/etc/ld.so.preload", R\_OK) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC) = 3

fstat(3, {st\_mode=S\_IFREG|0644, st\_size=19871, ...}) = 0

mmap(NULL, 19871, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7f841c404000

close(3) = 0

openat(AT\_FDCWD, "/lib/x86\_64-linux-gnu/libc.so.6", O\_RDONLY|O\_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\220\243\2\0\0\0\0\0"..., 832) = 832

pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784

fstat(3, {st\_mode=S\_IFREG|0755, st\_size=2125328, ...}) = 0

pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784

mmap(NULL, 2170256, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f841c1f2000

mmap(0x7f841c21a000, 1605632, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000) = 0x7f841c21a000

mmap(0x7f841c3a2000, 323584, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1b0000) = 0x7f841c3a2000

mmap(0x7f841c3f1000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1fe000) = 0x7f841c3f1000

mmap(0x7f841c3f7000, 52624, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7f841c3f7000

close(3) = 0

mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7f841c1ef000

arch\_prctl(ARCH\_SET\_FS, 0x7f841c1ef740) = 0

set\_tid\_address(0x7f841c1efa10) = 5558

set\_robust\_list(0x7f841c1efa20, 24) = 0

rseq(0x7f841c1f0060, 0x20, 0, 0x53053053) = 0

mprotect(0x7f841c3f1000, 16384, PROT\_READ) = 0

mprotect(0x5631824d3000, 4096, PROT\_READ) = 0

mprotect(0x7f841c441000, 8192, PROT\_READ) = 0

prlimit64(0, RLIMIT\_STACK, NULL, {rlim\_cur=8192\*1024, rlim\_max=RLIM64\_INFINITY}) = 0

munmap(0x7f841c404000, 19871) = 0

write(1, "Enter the file name for child1: ", 32Enter the file name for child1: ) = 32

read(0, f1.txt

"f1.txt\n", 255) = 7

write(1, "Enter the file name for child2: ", 32Enter the file name for child2: ) = 32

read(0, f2.txt

"f2.txt\n", 255) = 7

openat(AT\_FDCWD, "f1.txt", O\_WRONLY|O\_CREAT|O\_TRUNC, 0666) = 3

close(3) = 0

openat(AT\_FDCWD, "f2.txt", O\_WRONLY|O\_CREAT|O\_TRUNC, 0666) = 3

close(3) = 0

openat(AT\_FDCWD, "/dev/shm/shm1", O\_RDWR|O\_CREAT|O\_NOFOLLOW|O\_CLOEXEC, 0666) = 3

openat(AT\_FDCWD, "/dev/shm/shm2", O\_RDWR|O\_CREAT|O\_NOFOLLOW|O\_CLOEXEC, 0666) = 4

ftruncate(3, 1024) = 0

ftruncate(4, 1024) = 0

mmap(NULL, 1024, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 3, 0) = 0x7f841c408000

mmap(NULL, 1024, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 4, 0) = 0x7f841c407000

openat(AT\_FDCWD, "/dev/shm/sem.sem1", O\_RDWR|O\_NOFOLLOW|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

getrandom("\x3c\x75\x83\x19\x96\x8c\x46\x70", 8, GRND\_NONBLOCK) = 8

newfstatat(AT\_FDCWD, "/dev/shm/sem.ummIwG", 0x7fff0b3c23a0, AT\_SYMLINK\_NOFOLLOW) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/dev/shm/sem.ummIwG", O\_RDWR|O\_CREAT|O\_EXCL|O\_NOFOLLOW|O\_CLOEXEC, 0666) = 5

write(5, "\1\0\0\0\0\0\0\0\200\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0", 32) = 32

mmap(NULL, 32, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 5, 0) = 0x7f841c406000

link("/dev/shm/sem.ummIwG", "/dev/shm/sem.sem1") = 0

fstat(5, {st\_mode=S\_IFREG|0664, st\_size=32, ...}) = 0

getrandom("\x92\x56\x32\x9a\x6f\xb8\x3b\x59", 8, GRND\_NONBLOCK) = 8

brk(NULL) = 0x56318445c000

brk(0x56318447d000) = 0x56318447d000

unlink("/dev/shm/sem.ummIwG") = 0

close(5) = 0

openat(AT\_FDCWD, "/dev/shm/sem.sem2", O\_RDWR|O\_NOFOLLOW|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

getrandom("\xdc\x19\xd3\xd4\x67\x10\x64\x83", 8, GRND\_NONBLOCK) = 8

newfstatat(AT\_FDCWD, "/dev/shm/sem.uaqDnh", 0x7fff0b3c23a0, AT\_SYMLINK\_NOFOLLOW) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/dev/shm/sem.uaqDnh", O\_RDWR|O\_CREAT|O\_EXCL|O\_NOFOLLOW|O\_CLOEXEC, 0666) = 5

write(5, "\1\0\0\0\0\0\0\0\200\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0", 32) = 32

mmap(NULL, 32, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 5, 0) = 0x7f841c405000

link("/dev/shm/sem.uaqDnh", "/dev/shm/sem.sem2") = 0

fstat(5, {st\_mode=S\_IFREG|0664, st\_size=32, ...}) = 0

unlink("/dev/shm/sem.uaqDnh") = 0

close(5) = 0

clone(child\_stack=NULL, flags=CLONE\_CHILD\_CLEARTID|CLONE\_CHILD\_SETTID|SIGCHLDstrace: Process 5559 attached

, child\_tidptr=0x7f841c1efa10) = 5559

[pid 5558] clone(child\_stack=NULL, flags=CLONE\_CHILD\_CLEARTID|CLONE\_CHILD\_SETTID|SIGCHLD <unfinished ...>

[pid 5559] set\_robust\_list(0x7f841c1efa20, 24) = 0

strace: Process 5560 attached

[pid 5558] <... clone resumed>, child\_tidptr=0x7f841c1efa10) = 5560

[pid 5559] execve("./child.out", ["./child.out", "f1.txt", "/shm1", "/sem1"], 0x7fff0b3c2bf8 /\* 22 vars \*/ <unfinished ...>

[pid 5558] read(0, <unfinished ...>

[pid 5560] set\_robust\_list(0x7f841c1efa20, 24) = 0

[pid 5559] <... execve resumed>) = 0

[pid 5560] execve("./child.out", ["./child.out", "f2.txt", "/shm2", "/sem2"], 0x7fff0b3c2bf8 /\* 22 vars \*/ <unfinished ...>

[pid 5559] brk(NULL) = 0x55725574f000

[pid 5560] <... execve resumed>) = 0

[pid 5559] mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0 <unfinished ...>

[pid 5560] brk(NULL <unfinished ...>

[pid 5559] <... mmap resumed>) = 0x7f7c007a0000

[pid 5560] <... brk resumed>) = 0x55a344723000

[pid 5559] access("/etc/ld.so.preload", R\_OK <unfinished ...>

[pid 5560] mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0 <unfinished ...>

[pid 5559] <... access resumed>) = -1 ENOENT (No such file or directory)

[pid 5560] <... mmap resumed>) = 0x7f8b5e97a000

[pid 5559] openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC <unfinished ...>

[pid 5560] access("/etc/ld.so.preload", R\_OK <unfinished ...>

[pid 5559] <... openat resumed>) = 3

[pid 5560] <... access resumed>) = -1 ENOENT (No such file or directory)

[pid 5559] fstat(3, <unfinished ...>

[pid 5560] openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC <unfinished ...>

[pid 5559] <... fstat resumed>{st\_mode=S\_IFREG|0644, st\_size=19871, ...}) = 0

[pid 5560] <... openat resumed>) = 3

[pid 5560] fstat(3, <unfinished ...>

[pid 5559] mmap(NULL, 19871, PROT\_READ, MAP\_PRIVATE, 3, 0 <unfinished ...>

[pid 5560] <... fstat resumed>{st\_mode=S\_IFREG|0644, st\_size=19871, ...}) = 0

[pid 5560] mmap(NULL, 19871, PROT\_READ, MAP\_PRIVATE, 3, 0 <unfinished ...>

[pid 5559] <... mmap resumed>) = 0x7f7c0079b000

[pid 5560] <... mmap resumed>) = 0x7f8b5e975000

[pid 5559] close(3 <unfinished ...>

[pid 5560] close(3 <unfinished ...>

[pid 5559] <... close resumed>) = 0

[pid 5560] <... close resumed>) = 0

[pid 5559] openat(AT\_FDCWD, "/lib/x86\_64-linux-gnu/libc.so.6", O\_RDONLY|O\_CLOEXEC <unfinished ...>

[pid 5560] openat(AT\_FDCWD, "/lib/x86\_64-linux-gnu/libc.so.6", O\_RDONLY|O\_CLOEXEC <unfinished ...>

[pid 5559] <... openat resumed>) = 3

[pid 5560] <... openat resumed>) = 3

[pid 5559] read(3, <unfinished ...>

[pid 5560] read(3, <unfinished ...>

[pid 5559] <... read resumed>"\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\220\243\2\0\0\0\0\0"..., 832) = 832

[pid 5559] pread64(3, <unfinished ...>

[pid 5560] <... read resumed>"\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\220\243\2\0\0\0\0\0"..., 832) = 832

[pid 5559] <... pread64 resumed>"\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784

[pid 5560] pread64(3, <unfinished ...>

[pid 5559] fstat(3, <unfinished ...>

[pid 5560] <... pread64 resumed>"\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784

[pid 5559] <... fstat resumed>{st\_mode=S\_IFREG|0755, st\_size=2125328, ...}) = 0

[pid 5560] fstat(3, <unfinished ...>

[pid 5559] pread64(3, <unfinished ...>

[pid 5560] <... fstat resumed>{st\_mode=S\_IFREG|0755, st\_size=2125328, ...}) = 0

[pid 5560] pread64(3, <unfinished ...>

[pid 5559] <... pread64 resumed>"\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784

[pid 5560] <... pread64 resumed>"\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784

[pid 5559] mmap(NULL, 2170256, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0 <unfinished ...>

[pid 5560] mmap(NULL, 2170256, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0 <unfinished ...>

[pid 5559] <... mmap resumed>) = 0x7f7c00589000

[pid 5560] <... mmap resumed>) = 0x7f8b5e763000

[pid 5559] mmap(0x7f7c005b1000, 1605632, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000 <unfinished ...>

[pid 5560] mmap(0x7f8b5e78b000, 1605632, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000 <unfinished ...>

[pid 5559] <... mmap resumed>) = 0x7f7c005b1000

[pid 5560] <... mmap resumed>) = 0x7f8b5e78b000

[pid 5559] mmap(0x7f7c00739000, 323584, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1b0000 <unfinished ...>

[pid 5560] mmap(0x7f8b5e913000, 323584, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1b0000 <unfinished ...>

[pid 5559] <... mmap resumed>) = 0x7f7c00739000

[pid 5560] <... mmap resumed>) = 0x7f8b5e913000

[pid 5559] mmap(0x7f7c00788000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1fe000 <unfinished ...>

[pid 5560] mmap(0x7f8b5e962000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1fe000 <unfinished ...>

[pid 5559] <... mmap resumed>) = 0x7f7c00788000

[pid 5560] <... mmap resumed>) = 0x7f8b5e962000

[pid 5559] mmap(0x7f7c0078e000, 52624, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0 <unfinished ...>

[pid 5560] mmap(0x7f8b5e968000, 52624, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0 <unfinished ...>

[pid 5559] <... mmap resumed>) = 0x7f7c0078e000

[pid 5560] <... mmap resumed>) = 0x7f8b5e968000

[pid 5559] close(3 <unfinished ...>

[pid 5560] close(3 <unfinished ...>

[pid 5559] <... close resumed>) = 0

[pid 5560] <... close resumed>) = 0

[pid 5559] mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0 <unfinished ...>

[pid 5560] mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0 <unfinished ...>

[pid 5559] <... mmap resumed>) = 0x7f7c00586000

[pid 5560] <... mmap resumed>) = 0x7f8b5e760000

[pid 5559] arch\_prctl(ARCH\_SET\_FS, 0x7f7c00586740 <unfinished ...>

[pid 5560] arch\_prctl(ARCH\_SET\_FS, 0x7f8b5e760740 <unfinished ...>

[pid 5559] <... arch\_prctl resumed>) = 0

[pid 5560] <... arch\_prctl resumed>) = 0

[pid 5559] set\_tid\_address(0x7f7c00586a10 <unfinished ...>

[pid 5560] set\_tid\_address(0x7f8b5e760a10 <unfinished ...>

[pid 5559] <... set\_tid\_address resumed>) = 5559

[pid 5560] <... set\_tid\_address resumed>) = 5560

[pid 5560] set\_robust\_list(0x7f8b5e760a20, 24) = 0

[pid 5559] set\_robust\_list(0x7f7c00586a20, 24 <unfinished ...>

[pid 5560] rseq(0x7f8b5e761060, 0x20, 0, 0x53053053 <unfinished ...>

[pid 5559] <... set\_robust\_list resumed>) = 0

[pid 5560] <... rseq resumed>) = 0

[pid 5559] rseq(0x7f7c00587060, 0x20, 0, 0x53053053 <unfinished ...>

[pid 5560] mprotect(0x7f8b5e962000, 16384, PROT\_READ <unfinished ...>

[pid 5559] <... rseq resumed>) = 0

[pid 5560] <... mprotect resumed>) = 0

[pid 5560] mprotect(0x55a342861000, 4096, PROT\_READ <unfinished ...>

[pid 5559] mprotect(0x7f7c00788000, 16384, PROT\_READ <unfinished ...>

[pid 5560] <... mprotect resumed>) = 0

[pid 5559] <... mprotect resumed>) = 0

[pid 5560] mprotect(0x7f8b5e9b2000, 8192, PROT\_READ <unfinished ...>

[pid 5559] mprotect(0x557254421000, 4096, PROT\_READ <unfinished ...>

[pid 5560] <... mprotect resumed>) = 0

[pid 5559] <... mprotect resumed>) = 0

[pid 5560] prlimit64(0, RLIMIT\_STACK, NULL, <unfinished ...>

[pid 5559] mprotect(0x7f7c007d8000, 8192, PROT\_READ <unfinished ...>

[pid 5560] <... prlimit64 resumed>{rlim\_cur=8192\*1024, rlim\_max=RLIM64\_INFINITY}) = 0

[pid 5559] <... mprotect resumed>) = 0

[pid 5560] munmap(0x7f8b5e975000, 19871 <unfinished ...>

[pid 5559] prlimit64(0, RLIMIT\_STACK, NULL, <unfinished ...>

[pid 5560] <... munmap resumed>) = 0

[pid 5559] <... prlimit64 resumed>{rlim\_cur=8192\*1024, rlim\_max=RLIM64\_INFINITY}) = 0

[pid 5560] openat(AT\_FDCWD, "f2.txt", O\_WRONLY|O\_CREAT|O\_TRUNC, 0666 <unfinished ...>

[pid 5559] munmap(0x7f7c0079b000, 19871) = 0

[pid 5560] <... openat resumed>) = 3

[pid 5559] openat(AT\_FDCWD, "f1.txt", O\_WRONLY|O\_CREAT|O\_TRUNC, 0666 <unfinished ...>

[pid 5560] openat(AT\_FDCWD, "/dev/shm/shm2", O\_RDWR|O\_NOFOLLOW|O\_CLOEXEC <unfinished ...>

[pid 5559] <... openat resumed>) = 3

[pid 5560] <... openat resumed>) = 4

[pid 5559] openat(AT\_FDCWD, "/dev/shm/shm1", O\_RDWR|O\_NOFOLLOW|O\_CLOEXEC <unfinished ...>

[pid 5560] mmap(NULL, 1024, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 4, 0 <unfinished ...>

[pid 5559] <... openat resumed>) = 4

[pid 5560] <... mmap resumed>) = 0x7f8b5e979000

[pid 5559] mmap(NULL, 1024, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 4, 0 <unfinished ...>

[pid 5560] openat(AT\_FDCWD, "/dev/shm/sem.sem2", O\_RDWR|O\_NOFOLLOW|O\_CLOEXEC <unfinished ...>

[pid 5559] <... mmap resumed>) = 0x7f7c0079f000

[pid 5560] <... openat resumed>) = 5

[pid 5560] fstat(5, <unfinished ...>

[pid 5559] openat(AT\_FDCWD, "/dev/shm/sem.sem1", O\_RDWR|O\_NOFOLLOW|O\_CLOEXEC <unfinished ...>

[pid 5560] <... fstat resumed>{st\_mode=S\_IFREG|0664, st\_size=32, ...}) = 0

[pid 5559] <... openat resumed>) = 5

[pid 5560] getrandom( <unfinished ...>

[pid 5559] fstat(5, <unfinished ...>

[pid 5560] <... getrandom resumed>"\x24\x3c\x8b\x91\xcd\x6f\xf1\x81", 8, GRND\_NONBLOCK) = 8

[pid 5559] <... fstat resumed>{st\_mode=S\_IFREG|0664, st\_size=32, ...}) = 0

[pid 5560] brk(NULL <unfinished ...>

[pid 5559] getrandom( <unfinished ...>

[pid 5560] <... brk resumed>) = 0x55a344723000

[pid 5559] <... getrandom resumed>"\x68\xa3\xb7\xab\x19\x58\x95\x39", 8, GRND\_NONBLOCK) = 8

[pid 5560] brk(0x55a344744000) = 0x55a344744000

[pid 5559] brk(NULL <unfinished ...>

[pid 5560] mmap(NULL, 32, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 5, 0 <unfinished ...>

[pid 5559] <... brk resumed>) = 0x55725574f000

[pid 5560] <... mmap resumed>) = 0x7f8b5e978000

[pid 5559] brk(0x557255770000 <unfinished ...>

[pid 5560] close(5 <unfinished ...>

[pid 5559] <... brk resumed>) = 0x557255770000

[pid 5560] <... close resumed>) = 0

[pid 5559] mmap(NULL, 32, PROT\_READ|PROT\_WRITE, MAP\_SHARED, 5, 0 <unfinished ...>

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] <... mmap resumed>) = 0x7f7c0079e000

[pid 5559] close(5) = 0

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, w <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, qdNULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, d <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, wNULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, q <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, d <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000},

<unfinished ...>

[pid 5558] <... read resumed>"wqddwqd\n", 255) = 8

[pid 5558] read(0, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] write(3, "wqddwqd", 7) = 7

[pid 5560] write(3, "\n", 1) = 1

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, e <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, NULL) = 0

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, x <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, i <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, t <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000}, <unfinished ...>

[pid 5559] clock\_nanosleep(CLOCK\_REALTIME, 0, {tv\_sec=0, tv\_nsec=100000000},

<unfinished ...>

[pid 5558] <... read resumed>"exit\n", 255) = 5

[pid 5558] wait4(5559, <unfinished ...>

[pid 5560] <... clock\_nanosleep resumed>NULL) = 0

[pid 5559] <... clock\_nanosleep resumed>NULL) = 0

[pid 5560] close(3 <unfinished ...>

[pid 5559] close(3) = 0

[pid 5559] munmap(0x7f7c0079f000, 1024 <unfinished ...>

[pid 5560] <... close resumed>) = 0

[pid 5559] <... munmap resumed>) = 0

[pid 5560] munmap(0x7f8b5e979000, 1024 <unfinished ...>

[pid 5559] munmap(0x7f7c0079e000, 32 <unfinished ...>

[pid 5560] <... munmap resumed>) = 0

[pid 5560] munmap(0x7f8b5e978000, 32 <unfinished ...>

[pid 5559] <... munmap resumed>) = 0

[pid 5560] <... munmap resumed>) = 0

[pid 5560] exit\_group(0 <unfinished ...>

[pid 5559] exit\_group(0 <unfinished ...>

[pid 5560] <... exit\_group resumed>) = ?

[pid 5559] <... exit\_group resumed>) = ?

[pid 5560] +++ exited with 0 +++

[pid 5558] <... wait4 resumed>NULL, 0, NULL) = ? ERESTARTSYS (To be restarted if SA\_RESTART is set)

[pid 5559] +++ exited with 0 +++

--- SIGCHLD {si\_signo=SIGCHLD, si\_code=CLD\_EXITED, si\_pid=5560, si\_uid=1000, si\_status=0, si\_utime=0, si\_stime=0} ---

wait4(5559, NULL, 0, NULL) = 5559

wait4(5560, NULL, 0, NULL) = 5560

munmap(0x7f841c408000, 1024) = 0

munmap(0x7f841c407000, 1024) = 0

unlink("/dev/shm/shm1") = 0

unlink("/dev/shm/shm2") = 0

munmap(0x7f841c406000, 32) = 0

munmap(0x7f841c405000, 32) = 0

unlink("/dev/shm/sem.sem1") = 0

unlink("/dev/shm/sem.sem2") = 0

exit\_group(0) = ?

+++ exited with 0 +++

# Вывод

В процессе выполнения лабораторной работы я освоил основные принципы работы с разделяемой памятью на языке C. Я научился создавать объекты разделяемой памяти, записывать в них данные и считывать информацию из них. Кроме того, я познакомился с использованием семафоров для синхронизации процессов при доступе к разделяемой памяти, что позволило избежать конфликтов и обеспечить корректное взаимодействие между ними. Также я получил представление о работе файловых систем и общих принципах управления памятью в операционных системах.