Московский Авиационный Институт

(Национальный Исследовательский Университет)

Институт №8 “Компьютерные науки и прикладная математика”

Кафедра №806 “Вычислительная математика и программирование”

Лабораторная работа №1 по курсу

«Операционные системы»

Группа: М8О-213Б-23

Студент: Иванов В. М.

Преподаватель: Бахарев В.Д.

Оценка: _____

Дата: 18.10.24

Москва, 2024

Постановка задачи

Вариант 20.

Родительский процесс создает два дочерних процесса. Первой строкой пользователь в консоль родительского процесса вводит имя файла, которое будет использовано для открытия File с таким именем на запись для child1. Аналогично для второй строки и процесса child2. Родительский и дочерний процесс должны быть представлены разными программами. Родительский процесс принимает от пользователя строки произвольной длины и пересылает их в pipe1 или в pipe2 в зависимости от правила фильтрации. Процесс child1 и child2 производят работу над строками. Процессы пишут результаты своей работы в стандартный вывод.

Вариант 20) Правило фильтрации: строки длины больше 10 символов отправляются в pipe2, иначе в pipe1. Дочерние процессы инвертируют строки.

Общий метод и алгоритм решения

Кратко опишите системные вызовы, которые вы использовали в лабораторной работе.

Использованные системные вызовы:

- pid_t fork(void); – создает дочерний процесс.
- int pipe(int *fd); – создает неименованный канал для передачи данных между
- процессами.
- void exit(int status) — завершение выполнения процесса и возвращение статуса.
- int dup2(int oldfd, int newfd) — переназначение файлового дескриптора.
- int close(int fd) — закрыть файл.
- int execl() - заменяет текущий процесс на новый процесс, загружая исполняемый файл.
- int open() - открытие/создание файла.
- int write() - вывод на экран сообщение.
- int read() - чтение с файла.

Общий алгоритм:

- Запросить у пользователя названия файлов
- Создать каналы
- форкнуть процесс и переназначить stdin на pipeN[0] а stdout на открытый файл
- запустить дочерний процесс через execl()
- запрашивать у пользователя строки и в зависимости от длины строки направляем в разные pipe`ы

Код программы

main.c

```
#include <unistd.h>

#include <string.h>

#include <fcntl.h>

#include <stdlib.h>


#define BUFSIZ 1024



#define ECPIPE "Error creating pipe\n"
```

```
#define ECHILD1 "Error creating process 1\n"
#define EFCHILD1 "Error opening file for child1\n"
#define EECHILD1 "Exec error for child1\n"
#define ECHILD2 "Error creating process 2\n"
#define EFCHILD2 "Error opening file for child2\n"
#define EECHILD2 "Exec error for child2\n"
#define ERLINE "Error reading line\n"


int main() {
    int pipe1[2], pipe2[2];
    char buffer[BUFSIZ];
    int count = 1;
    char filename1[BUFSIZ];
    char filename2[BUFSIZ];

    if (pipe(pipe1) == -1 || pipe(pipe2) == -1) {
        write(2, ECPIPE, strlen(ECPIPE));
        exit(EXIT_FAILURE);
    }

    write(1, "Enter a filename for child1: ", 30);
    read(0, filename1, BUFSIZ);
    filename1[strcspn(filename1, "\n")] = '\0';

    write(1, "Enter a filename for child2: ", 30);
    read(0, filename2, BUFSIZ);
    filename2[strcspn(filename2, "\n")] = '\0';


    pid_t pid1 = fork();
    if (pid1 == -1) {
        write(2, ECHILD1, strlen(ECHILD1));
        exit(EXIT_FAILURE);
    }
```

```
if (pid1 == 0) {
    int fd1 = open(filename1, O_WRONLY | O_CREAT | O_TRUNC, 0644);
    if (fd1 == -1) {
        write(2, EFCHILD1, strlen(EFCHILD1));
        exit(EXIT_FAILURE);
    }

    close(pipe1[1]);
    dup2(pipe1[0], 0);
    dup2(fd1, 1);
    close(pipe1[0]);
    close(fd1);

    execl("./child", "child", NULL);
    write(2, "exec error for child1.\n", 24);
    exit(EXIT_FAILURE);
}



pid_t pid2 = fork();
if (pid2 == -1) {
    write(2, ECHILD2, strlen(ECHILD2));
    exit(EXIT_FAILURE);
}

if (pid2 == 0) {
    int fd2 = open(filename2, O_WRONLY | O_CREAT | O_TRUNC, 0644);
    if (fd2 == -1) {
        write(2, EFCHILD2, strlen(EFCHILD2));
        exit(EXIT_FAILURE);
    }

    close(pipe2[1]);
    dup2(pipe2[0], 0);
    dup2(fd2, 1);
    close(pipe2[0]);
```

```
        close(fd2);

        execl("./child", "child", NULL);
        write(2, EECHILD2, strlen(EECHILD2));
        exit(EXIT_FAILURE);
    }
    close(pipe1[0]);
    close(pipe2[0]);

    while (1) {
        write(1, "Enter line: ", 13);
        ssize_t bytes_read = read(0, buffer, BUFSIZ);
        if (bytes_read == 0) { // EOF
            write(pipe2[1], "\0", 1);
            write(pipe1[1], "\0", 1);
            exit(EXIT_SUCCESS);
        } else if (bytes_read == -1){
            write(2, ERLINE, strlen(ERLINE));
            exit(EXIT_FAILURE);
        }
        buffer[bytes_read - 1] = '\0';

        if (strlen(buffer) == 0) {
            break;
        }

        if (strlen(buffer) > 10) {
            write(pipe2[1], buffer, bytes_read);
        } else {
            write(pipe1[1], buffer, bytes_read);
        }
    }
    close(pipe1[1]);
    close(pipe2[1]);
    return 0;
}
```

child.c

```
#include <unistd.h>
#include <string.h>
#include <stdlib.h>

#define BUFSIZ 1024

void reverse_string(char *str) {
    int len = strlen(str);
    for (int i = 0; i < len / 2; i++) {
        char tmp = str[i];
        str[i] = str[len - 1 - i];
        str[len - 1 - i] = tmp;
    }
}

int main() {
    char buf[BUFSIZ];

    while (1) {
        ssize_t readed = read(0, buf, BUFSIZ);
        if (readed <= 0) { //
            exit(EXIT_SUCCESS);
        }
        buf[readed - 1] = '\0';

        if (strlen(buf) == 0) {
            break;
        }

        reverse_string(buf);
        write(1, buf, strlen(buf));
        write(1, "\n", 1);
    }
```

```
    return 0;
}
```

Протокол работы программы

Тестирование:

```
$ ./main
Enter a filename for child1: out1.txt
Enter a filename for child2: out2.txt
Enter line: asd
Enter line: 123
Enter line: 12345678901
Enter line: asdfghjklqw
Enter line:
$ cat out1.txt
dsa
321
$ cat out2.txt
10987654321
wqlkjhgfdsa
```

Strace:

```
    196727 execve("./main", ["./main"], 0x7fffe57369c8 /* 98 vars */) = 0
    196727 brk(NULL)                        = 0x21da000
    196727 arch_prctl(0x3001 /* ARCH_??? */, 0x7ffff6d87200) = -1 EINVAL (Недопустимый аргумент)
    196727 access("/etc/ld.so.preload", R_OK) = -1 ENOENT (Нет такого файла или каталога)
    196727 openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
    196727 fstat(3, {st_mode=S_IFREG|0644, st_size=95083, ...}) = 0
    196727 mmap(NULL, 95083, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f81f1486000
    196727 close(3)                         = 0
    196727 openat(AT_FDCWD, "/lib64/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
    196727      read(3,      "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\227\2\0\0\0\0\0"..., 832) = 832
    196727     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
```

```
196727           pread64(3,           "\4\0\0\0          \0\0\0\5\0\0\0GNU\
0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0"..., 48, 848) = 48

196727             pread64(3,             "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0T\
247\253\1\356\366\342\334\242\306\260\332\270\306V\241"..., 68, 896) = 68

196727 fstat(3, {st_mode=S_IFREG|0755, st_size=2592552, ...}) = 0

196727 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -
1, 0) = 0x7f81f1484000

196727     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

196727 mmap(NULL, 2133936, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f81f1200000

196727 mprotect(0x7f81f1228000, 1892352, PROT_NONE) = 0

196727  mmap(0x7f81f1228000,  1527808,  PROT_READ|PROT_EXEC,  MAP_PRIVATE|
MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7f81f1228000

196727  mmap(0x7f81f139d000,  360448,  PROT_READ,  MAP_PRIVATE|MAP_FIXED|
MAP_DENYWRITE, 3, 0x19d000) = 0x7f81f139d000

196727  mmap(0x7f81f13f6000,  24576,  PROT_READ|PROT_WRITE,  MAP_PRIVATE|
MAP_FIXED|MAP_DENYWRITE, 3, 0x1f5000) = 0x7f81f13f6000

196727  mmap(0x7f81f13fc000,  53168,  PROT_READ|PROT_WRITE,  MAP_PRIVATE|
MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f81f13fc000

196727 close(3)                         = 0

196727 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -
1, 0) = 0x7f81f1482000

196727 arch_prctl(ARCH_SET_FS, 0x7f81f1485600) = 0

196727 set_tid_address(0x7f81f14858d0)  = 196727

196727 set_robust_list(0x7f81f14858e0, 24) = 0

196727 rseq(0x7f81f1485fa0, 0x20, 0, 0x53053053) = 0

196727 mprotect(0x7f81f13f6000, 16384, PROT_READ) = 0

196727 mprotect(0x403000, 4096, PROT_READ) = 0

196727 mprotect(0x7f81f14d2000, 8192, PROT_READ) = 0

196727    prlimit64(0,    RLIMIT_STACK,    NULL,    {rlim_cur=8192*1024,
rlim_max=RLIM64_INFINITY}) = 0

196727 munmap(0x7f81f1486000, 95083)    = 0

196727 pipe([3, 4])                     = 0

196727 pipe([5, 6])                     = 0

196727 write(1, "Enter a filename for child1: \0", 30) = 30

196727 read(0, "out1.txt\n", 1024)      = 9

196727 write(1, "Enter a filename for child2: \0", 30) = 30

196727 read(0, "out2.txt\n", 1024)      = 9

196727         clone(child_stack=NULL,         flags=CLONE_CHILD_CLEARTID|
CLONE_CHILD_SETTID|SIGCHLD, child_tidptr=0x7f81f14858d0) = 196822
```

```
196822 set_robust_list(0x7f81f14858e0, 24 <unfinished ...>
196727 clone(child_stack=NULL, flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLD <unfinished ...>
196822 <... set_robust_list resumed>)   = 0
196727 <... clone resumed>, child_tidptr=0x7f81f14858d0) = 196823
196823 set_robust_list(0x7f81f14858e0, 24 <unfinished ...>
196727 close(3 <unfinished ...>
196822 openat(AT_FDCWD, "out1.txt", O_WRONLY|O_CREAT|O_TRUNC, 0644 <unfinished ...>
196727 <... close resumed>)             = 0
196823 <... set_robust_list resumed>)   = 0
196727 close(5)                         = 0
196727 write(1, "Enter line: \0", 13 <unfinished ...>
196823 openat(AT_FDCWD, "out2.txt", O_WRONLY|O_CREAT|O_TRUNC, 0644 <unfinished ...>
196822 <... openat resumed>)            = 7
196727 <... write resumed>)             = 13
196822 close(4 <unfinished ...>
196727 read(0,  <unfinished ...>
196822 <... close resumed>)             = 0
196823 <... openat resumed>)            = 7
196822 dup2(3, 0 <unfinished ...>
196823 close(6 <unfinished ...>
196822 <... dup2 resumed>)              = 0
196823 <... close resumed>)             = 0
196822 dup2(7, 1 <unfinished ...>
196823 dup2(5, 0 <unfinished ...>
196822 <... dup2 resumed>)              = 1
196823 <... dup2 resumed>)              = 0
196822 close(3 <unfinished ...>
196823 dup2(7, 1 <unfinished ...>
196822 <... close resumed>)             = 0
196823 <... dup2 resumed>)              = 1
196822 close(7 <unfinished ...>
196823 close(5 <unfinished ...>
196822 <... close resumed>)             = 0
196823 <... close resumed>)             = 0
```

```
196822  execve("./child", ["child"], 0x7ffff6d87328 /* 98 vars */
<unfinished ...>
196823 close(7)                          = 0
196823  execve("./child", ["child"], 0x7ffff6d87328 /* 98 vars */
<unfinished ...>
196822 <... execve resumed>)            = 0
196822 brk(NULL)                        = 0x53d000
196823 <... execve resumed>)            = 0
196822 arch_prctl(0x3001 /* ARCH_??? */, 0x7fffe179f510 <unfinished ...>
196823 brk(NULL <unfinished ...>
196822 <... arch_prctl resumed>)             = -1 EINVAL (Недопустимый
аргумент)
196823 <... brk resumed>)               = 0x13e2000
196822 access("/etc/ld.so.preload", R_OK <unfinished ...>
196823 arch_prctl(0x3001 /* ARCH_??? */, 0x7ffc33a6a1f0) = -1 EINVAL
(Недопустимый аргумент)
196822 <... access resumed>)             = -1 ENOENT (Нет такого файла или
каталога)
196823 access("/etc/ld.so.preload", R_OK <unfinished ...>
196822 openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC <unfinished
...>
196823 <... access resumed>)             = -1 ENOENT (Нет такого файла или
каталога)
196822 <... openat resumed>)            = 3
196823 openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC <unfinished
...>
196822 fstat(3,  <unfinished ...>
196823 <... openat resumed>)            = 5
196822 <... fstat resumed>{st_mode=S_IFREG|0644, st_size=95083, ...}) = 0
196822 mmap(NULL, 95083, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f69fd564000
196822 close(3)                         = 0
196823 fstat(5,  <unfinished ...>
196822 openat(AT_FDCWD, "/lib64/libc.so.6", O_RDONLY|O_CLOEXEC <unfinished
...>
196823 <... fstat resumed>{st_mode=S_IFREG|0644, st_size=95083, ...}) = 0
196823 mmap(NULL, 95083, PROT_READ, MAP_PRIVATE, 5, 0 <unfinished ...>
196822 <... openat resumed>)            = 3
196823 <... mmap resumed>)              = 0x7f7f76462000
196823 close(5 <unfinished ...>
196822 read(3,  <unfinished ...>
```

```
     196823 <... close resumed>)             = 0

     196822 <... read resumed>"\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\
227\2\0\0\0\0"..., 832) = 832

     196823 openat(AT_FDCWD, "/lib64/libc.so.6", O_RDONLY|O_CLOEXEC <unfinished
...>

     196822 pread64(3,  <unfinished ...>

     196823 <... openat resumed>)            = 5

     196823 read(5,  <unfinished ...>

     196822     <...     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

     196823 <... read resumed>"\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\
227\2\0\0\0\0"..., 832) = 832

     196822 pread64(3,  <unfinished ...>

     196823 pread64(5,  <unfinished ...>

     196822     <...     pread64     resumed>"\4\0\0\0      \0\0\0\5\0\0\0GNU\
0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48, 848) = 48

     196823     <...     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

     196822 pread64(3,  <unfinished ...>

     196823 pread64(5,  <unfinished ...>

     196822     <...     pread64     resumed>"\4\0\0\0\24\0\0\0\3\0\0\0GNU\0T\
247\253\1\356\366\342\334\242\306\260\332\270\306V\241"..., 68, 896) = 68

     196823     <...     pread64     resumed>"\4\0\0\0      \0\0\0\5\0\0\0GNU\
0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48, 848) = 48

     196822 fstat(3,  <unfinished ...>

     196823 pread64(5,  <unfinished ...>

     196822 <... fstat resumed>{st_mode=S_IFREG|0755, st_size=2592552, ...}) =
0

     196823     <...     pread64     resumed>"\4\0\0\0\24\0\0\0\3\0\0\0GNU\0T\
247\253\1\356\366\342\334\242\306\260\332\270\306V\241"..., 68, 896) = 68

     196822 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -
1, 0 <unfinished ...>

     196823 fstat(5,  <unfinished ...>

     196822 <... mmap resumed>)              = 0x7f69fd562000

     196823 <... fstat resumed>{st_mode=S_IFREG|0755, st_size=2592552, ...}) =
0

     196822 pread64(3,  <unfinished ...>

     196823 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -
1, 0 <unfinished ...>

     196822     <...     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

     196823 <... mmap resumed>)              = 0x7f7f76460000
```

```
196822 mmap(NULL, 2133936, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0 <unfinished ...>

196823 pread64(5, <unfinished ...>

196822 <... mmap resumed>)              = 0x7f69fd200000

196823     <...     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

196822 mprotect(0x7f69fd228000, 1892352, PROT_NONE <unfinished ...>

196823 mmap(NULL, 2133936, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 5, 0 <unfinished ...>

196822 <... mprotect resumed>)          = 0

196823 <... mmap resumed>)              = 0x7f7f76200000

196822 mmap(0x7f69fd228000, 1527808, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000 <unfinished ...>

196823 mprotect(0x7f7f76228000, 1892352, PROT_NONE <unfinished ...>

196822 <... mmap resumed>)              = 0x7f69fd228000

196823 <... mprotect resumed>)          = 0

196822 mmap(0x7f69fd39d000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x19d000 <unfinished ...>

196823 mmap(0x7f7f76228000, 1527808, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 5, 0x28000 <unfinished ...>

196822 <... mmap resumed>)              = 0x7f69fd39d000

196823 <... mmap resumed>)              = 0x7f7f76228000

196822 mmap(0x7f69fd3f6000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1f5000 <unfinished ...>

196823 mmap(0x7f7f7639d000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 5, 0x19d000 <unfinished ...>

196822 <... mmap resumed>)              = 0x7f69fd3f6000

196823 <... mmap resumed>)              = 0x7f7f7639d000

196822 mmap(0x7f69fd3fc000, 53168, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0 <unfinished ...>

196823 mmap(0x7f7f763f6000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 5, 0x1f5000 <unfinished ...>

196822 <... mmap resumed>)              = 0x7f69fd3fc000

196823 <... mmap resumed>)              = 0x7f7f763f6000

196823 mmap(0x7f7f763fc000, 53168, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0 <unfinished ...>

196822 close(3 <unfinished ...>

196823 <... mmap resumed>)              = 0x7f7f763fc000

196822 <... close resumed>)             = 0

196822 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0 <unfinished ...>
```

```
196823 close(5 <unfinished ...>
196822 <... mmap resumed>)              = 0x7f69fd560000
196823 <... close resumed>)             = 0
196822 arch_prctl(ARCH_SET_FS, 0x7f69fd563600 <unfinished ...>
196823 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -
1, 0 <unfinished ...>
196822 <... arch_prctl resumed>)        = 0
196823 <... mmap resumed>)              = 0x7f7f7645e000
196822 set_tid_address(0x7f69fd5638d0 <unfinished ...>
196823 arch_prctl(ARCH_SET_FS, 0x7f7f76461600 <unfinished ...>
196822 <... set_tid_address resumed>)   = 196822
196823 <... arch_prctl resumed>)        = 0
196822 set_robust_list(0x7f69fd5638e0, 24 <unfinished ...>
196823 set_tid_address(0x7f7f764618d0 <unfinished ...>
196822 <... set_robust_list resumed>)   = 0
196823 <... set_tid_address resumed>)   = 196823
196822 rseq(0x7f69fd563fa0, 0x20, 0, 0x53053053 <unfinished ...>
196823 set_robust_list(0x7f7f764618e0, 24 <unfinished ...>
196822 <... rseq resumed>)              = 0
196823 <... set_robust_list resumed>)   = 0
196823 rseq(0x7f7f76461fa0, 0x20, 0, 0x53053053 <unfinished ...>
196822 mprotect(0x7f69fd3f6000, 16384, PROT_READ <unfinished ...>
196823 <... rseq resumed>)              = 0
196822 <... mprotect resumed>)          = 0
196823 mprotect(0x7f7f763f6000, 16384, PROT_READ <unfinished ...>
196822 mprotect(0x403000, 4096, PROT_READ <unfinished ...>
196823 <... mprotect resumed>)          = 0
196822 <... mprotect resumed>)          = 0
196823 mprotect(0x403000, 4096, PROT_READ <unfinished ...>
196822 mprotect(0x7f69fd5b0000, 8192, PROT_READ <unfinished ...>
196823 <... mprotect resumed>)          = 0
196822 <... mprotect resumed>)          = 0
196823 mprotect(0x7f7f764ae000, 8192, PROT_READ) = 0
196822 prlimit64(0, RLIMIT_STACK, NULL,  <unfinished ...>
196823 prlimit64(0, RLIMIT_STACK, NULL,  <unfinished ...>
```

```
    196822         <...          prlimit64         resumed>{rlim_cur=8192*1024,
rlim_max=RLIM64_INFINITY}) = 0
    196823         <...          prlimit64         resumed>{rlim_cur=8192*1024,
rlim_max=RLIM64_INFINITY}) = 0
    196822 munmap(0x7f69fd564000, 95083 <unfinished ...>
    196823 munmap(0x7f7f76462000, 95083)    = 0
    196822 <... munmap resumed>)            = 0
    196823 read(0,  <unfinished ...>
    196822 read(0,  <unfinished ...>
    196727 <... read resumed>"asd\n", 1024) = 4
    196727 write(4, "asd\0", 4)             = 4
    196822 <... read resumed>"asd\0", 1024) = 4
    196727 write(1, "Enter line: \0", 13)   = 13
    196822 write(1, "dsa", 3 <unfinished ...>
    196727 read(0,  <unfinished ...>
    196822 <... write resumed>)             = 3
    196822 write(1, "\n", 1)                = 1
    196822 read(0,  <unfinished ...>
    196727 <... read resumed>"123\n", 1024) = 4
    196727 write(4, "123\0", 4)             = 4
    196822 <... read resumed>"123\0", 1024) = 4
    196727 write(1, "Enter line: \0", 13)   = 13
    196822 write(1, "321", 3 <unfinished ...>
    196727 read(0,  <unfinished ...>
    196822 <... write resumed>)             = 3
    196822 write(1, "\n", 1)                = 1
    196822 read(0,  <unfinished ...>
    196727 <... read resumed>"qwertyuiop1\n", 1024) = 12
    196727 write(6, "qwertyuiop1\0", 12)    = 12
    196823 <... read resumed>"qwertyuiop1\0", 1024) = 12
    196727 write(1, "Enter line: \0", 13)   = 13
    196823 write(1, "1poiuytrewq", 11 <unfinished ...>
    196727 read(0,  <unfinished ...>
    196823 <... write resumed>)             = 11
    196823 write(1, "\n", 1)                = 1
    196823 read(0,  <unfinished ...>
```

```
196727 <... read resumed>"12345678901\n", 1024) = 12
196727 write(6, "12345678901\0", 12)    = 12
196823 <... read resumed>"12345678901\0", 1024) = 12
196727 write(1, "Enter line: \0", 13)   = 13
196823 write(1, "10987654321", 11 <unfinished ...>
196727 read(0,  <unfinished ...>
196823 <... write resumed>)             = 11
196823 write(1, "\n", 1)                = 1
196823 read(0,  <unfinished ...>
196727 <... read resumed>"", 1024)      = 0
196727 write(6, "\0", 1)                = 1
196823 <... read resumed>"\0", 1024)    = 1
196727 write(4, "\0", 1)                = 1
196823 exit_group(0 <unfinished ...>
196822 <... read resumed>"\0", 1024)    = 1
196727 exit_group(0)                    = ?
196823 <... exit_group resumed>)        = ?
196727 +++ exited with 0 +++
196823 +++ exited with 0 +++
196822 exit_group(0)                    = ?
196822 +++ exited with 0 +++
```

Вывод

Я научился созадваьб процессы blinux с помощщью системных вызовов. Научился открывать неименованные каналы и передавать данные по ним. Так же я использовал dup2() для переопределения файловых дескрипторов. Эти знания помогут мне лучше разобраться в принципах написания низкоуровнего системного ПО и в устройстве операционных систем.