

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-209БВ-24

Студент: Крюков Д.М.

Преподаватель: Миронов Е.С.

Оценка: \_\_\_\_\_

Дата: 06.10.25

Москва, 2025

# Постановка задачи

## Вариант 13.

Родительский процесс создает два дочерних процесса. Перенаправление стандартных потоков ввода-вывода показано на картинке выше. Child1 и Child2 можно «соединить» между собой дополнительным каналом. Родительский и дочерний процесс должны быть представлены разными программами. Родительский процесс принимает от пользователя строки произвольной длины и пересылает их в pipe1. Процесс child1 и child2 производят работу над строками. Child2 пересылает результат своей работы родительскому процессу. Родительский процесс полученный результат выводит в стандартный поток вывода.

## Общий метод и алгоритм решения

Использованные системные вызовы:

- `pid_t fork(void)` - создает дочерний процесс, который является копией родительского процесса.
- `int pipe(int *fd)` - создает канал и возвращает два дескриптора
- `int close(int fd)` - закрывает файловый дескриптор
- `int dup2(int oldfd, int newfd)` – перенаправляет стандартные потоки ввода-вывода
- `int execl(const char *path, const char *arg, ...)` - загружает и запускает новую программу, полностью заменяя текущий процесс
- `pid_t waitpid(pid_t pid, int* status, int options)` – ожидает завершения дочернего процесса.

Программа реализует многопроцессную обработку текстовых данных с использованием механизма каналов (pipes) для межпроцессного взаимодействия. Алгоритм начинается с создания трех каналов pipe, которые обеспечивают однонаправленную передачу данных между процессами. Затем родительский процесс создает два дочерних процесса с помощью системного вызова `fork`.

В первом дочернем процессе (child1) перенаправляются стандартные потоки ввода-вывода: ввод связывается с первым каналом, а вывод - со вторым каналом. После этого процесс child1 выполняет программу, которая преобразует полученные строки в нижний регистр. Во втором дочернем процессе (child2) аналогичным образом настраиваются потоки - ввод из второго канала и вывод в третий канал, после чего выполняется программа, заменяющая пробельные символы на подчеркивания.

Родительский процесс выполняет роль координатора: он читает строки из стандартного ввода, передает их через первый канал в child1, получает окончательный результат через третий канал от child2 и выводит обработанные строки пользователю. Все процессы корректно закрывают неиспользуемые файловые дескрипторы каналов для избежания утечек ресурсов. Программа работает в интерактивном режиме до получения сигнала завершения (Ctrl+D), после чего корректно завершает все процессы и освобождает ресурсы.

## Код программы

### parent.c

```
#include <stdio.h>
```

```

#include <stdlib.h>

#include <unistd.h>

#include <sys/wait.h>

#include <string.h>


#define BUFFER_SIZE 1024


int main() {

    int pipe1[2], pipe2[2], pipe3[2];


    printf("Parent процесс, PID: %d\n", getpid());


    if (pipe(pipe1) == -1 || pipe(pipe2) == -1 || pipe(pipe3) == -1) {

        perror("pipe failed");

        exit(EXIT_FAILURE);

    }


    pid_t pid1 = fork();

    if (pid1 == 0) {

        close(pipe1[1]); close(pipe2[0]); close(pipe3[0]); close(pipe3[1]);

        dup2(pipe1[0], STDIN_FILENO);

        dup2(pipe2[1], STDOUT_FILENO);

        execl("./child1", "child1", NULL);

        perror("execl child1 failed");

        exit(EXIT_FAILURE);

    }


    pid_t pid2 = fork();

    if (pid2 == 0) {

        close(pipe1[0]); close(pipe1[1]); close(pipe2[1]); close(pipe3[0]);

```

```

    dup2(pipe2[0], STDIN_FILENO);

    dup2(pipe3[1], STDOUT_FILENO);

    execl("./child2", "child2", NULL);

    perror("execl child2 failed");

    exit(EXIT_FAILURE);
}

close(pipe1[0]); close(pipe2[0]); close(pipe2[1]); close(pipe3[1]);

printf("Введите строку:\n");

char input[BUFFER_SIZE], output[BUFFER_SIZE];
ssize_t bytes_read;

while (fgets(input, BUFFER_SIZE, stdin) != NULL) {
    write(pipe1[1], input, strlen(input));

    bytes_read = read(pipe3[0], output, BUFFER_SIZE - 1);

    if (bytes_read > 0) {
        output[bytes_read] = '\0';

        printf("Итог: %s", output);
    }
}

close(pipe1[1]); close(pipe3[0]);

waitpid(pid1, NULL, 0); waitpid(pid2, NULL, 0);

printf("Parent завершен.\n");

return 0;
}

```

### **child1.c**

```
#include <stdio.h>
```

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

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

```
#include <ctype.h>
```

```
#define BUFFER_SIZE 1024
```

```
int main() {
```

```
    printf("Child1 запущен, PID: %d\n", getpid());
```

```
    char buffer[BUFFER_SIZE];
```

```
    ssize_t bytes_read;
```

```
    while ((bytes_read = read(STDIN_FILENO, buffer, BUFFER_SIZE)) > 0) {
```

```
        for (int i = 0; i < bytes_read; i++) buffer[i] = tolower(buffer[i]);
```

```
        write(STDOUT_FILENO, buffer, bytes_read);
```

```
    }
```

```
    printf("Child1 завершен\n");
```

```
    return 0;
```

```
}
```

### **child2.c**

```
#include <stdio.h>
```

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

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

```
#include <ctype.h>
```

```
#define BUFFER_SIZE 1024
```

```
int main() {
```

```

printf("Child2 запущен, PID: %d\n", getpid());

char buffer[BUFFER_SIZE];

ssize_t bytes_read;

while ((bytes_read = read(STDIN_FILENO, buffer, BUFFER_SIZE)) > 0) {

    for (int i = 0; i < bytes_read; i++)

        if (isspace(buffer[i]) && buffer[i] != '\n') buffer[i] = '_';

    write(STDOUT_FILENO, buffer, bytes_read);

}

printf("Child2 завершен\n");

return 0;

}

```

## Протокол работы программы

```

● root@01d40df164ff:/workspace/lab1/src# ./parent
Parent процесс, PID: 7315
Введите строку:
Hello World!
Итог: hello_world!
OAJDJAIDJ sdjaodiajsdoaj SAKDJAJ
Итог: oajdjaoidj_sdjaodiajsdoaj_sakdjaj
Parent завершен.

```

### Strace:

```

execve("./parent", ["./parent"], 0x7ffcdd5fa408 /* 27 vars */) = 0
brk(NULL)                               = 0x2ef27000
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7b3f7eda4000
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=29003, ...}) = 0
mmap(NULL, 29003, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7b3f7ed9c000

```

```
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\3\0>\0\1\0\0\0p\236\2\0\0\0\0\0@ \0\0\0\0\0\0\220\202\36\0
\0\0\0\0\0\0\0\0@\08\0\17\0@\0=\0< \0\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\0H\3\0\0\0\0\0\0\0H\3\0\0\0\0\0\0\0\10\0\0\0\0\0\0\0\3\0\0\0\4\0\0\0`$\36\0\0\0\0\0`$\36\0\0
\0\0\0`$\36\0\0\0\0\0\34\0\0\0\0\0\0\0\34\0\0\0\0\0\0\0\20\0\0\0\0\0\0\0\1\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\0\0\0\350t\2\0\0\0\0\0\0\350t\2\0\0\0\0\0\0\20\0\0\0\0\
\0\0\1\0\0\0\5\0\0\0\0\200\2\0\0\0\0\0\0\0\200\2\0\0\0\0\0\0\200\2\0\0\0\0\0\0}A\26\0\0\0\0\0}A\2
6\0\0\0\0\0\0\20\0\0\0\0\0\0\0\1\0\0\0\4\0\0\0\0\320\30\0\0\0\0\0\0\0\320\30\0\0\0\0\0\0\0\320\30\
\0\0\0\0\0|T\5\0\0\0\0\0\0|T\5\0\0\0\0\0\0\0\20\0\0\0\0\0\0\0\1\0\0\0\6\0\0\0x+\36\0\0\0\0\0x;\36\0
\0\0\0\0x;\36\0\0\0\0\0pk\0\0\0\0\0\0                                #\1\0\0\0\0\0\0\20\0\0\0\0\0\0\2\0\0\0\6\0\0\0
Y\36\0\0\0\0\0\0                                                    i\36\0\0\0\0\0\0
i\36\0\0\0\0\0\0@\2\0\0\0\0\0\0\0@\2\0\0\0\0\0\0\0\10\0\0\0\0\0\0\0\4\0\0\0\4\0\0\0\210\3\0\0\0\0\
\0\0\210\3\0\0\0\0\0\0\0\210\3\0\0\0\0\0\0\0 \0\0\0\0\0\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\0H\3\0\0\0\0\0\0\0H\3\0\0\0\0\0\0\0
\10\0\0\0\0\0\0\0\3\0\0\0\4\0\0\0`$\36\0\0\0\0\0`$\36\0\0\0\0\0`$\36\0\0\0\0\0\34\0\0\0\0\0\0\0
0\0\34\0\0\0\0\0\0\0\20\0\0\0\0\0\0\0\1\0\0\0\4\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0\0
0\0\0\0\0\350t\2\0\0\0\0\0\350t\2\0\0\0\0\0\0\20\0\0\0\0\0\0\1\0\0\0\5\0\0\0\0\200\2\0\0\0\0\0
\0\0\200\2\0\0\0\0\0\0\200\2\0\0\0\0\0\0}A\26\0\0\0\0\0\0}A\26\0\0\0\0\0\0\0\20\0\0\0\0\0\0\0\1\0\0\0
0\4\0\0\0\0\0\320\30\0\0\0\0\0\0\0\320\30\0\0\0\0\0\0\0\320\30\0\0\0\0\0\0\0\T\5\0\0\0\0\0\0\T\5\0\0\0\0\0
\0\0\20\0\0\0\0\0\0\0\1\0\0\0\6\0\0\0\0x+\36\0\0\0\0\0\0x;\36\0\0\0\0\0\0x;\36\0\0\0\0\0\0\0pk\0\0\0\0\0
\0      #\1\0\0\0\0\0\0\0\20\0\0\0\0\0\0\0\2\0\0\0\6\0\0\0      Y\36\0\0\0\0\0      i\36\0\0\0\0\0\0
i\36\0\0\0\0\0\0\0@\2\0\0\0\0\0\0\0@\2\0\0\0\0\0\0\0\10\0\0\0\0\0\0\0\0\4\0\0\0\4\0\0\0\210\3\0\0\0\0\0
0\0\210\3\0\0\0\0\0\0\0\210\3\0\0\0\0\0\0\0      \0\0\0\0\0\0\0\0
\0\0\0\0\0\0\0\10\0\0\0\0\0\0\0\4\0\0\0\4\0\0\0\250\3\0\0\0\0\0\0\250\3\0\0\0\0\0\0\250\3\0\0
0\0\0\0\0\0$\0\0\0\0\0\0\0$\0\0\0\0\0\0\0\4\0\0\0"... , 840, 64) = 840
```

```
fstat(3, {st_mode=S_IFREG|0755, st_size=2003408, ...}) = 0
```

```
pread64(3,
```

```
"\6\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\0H\3\0\0\0\0\0\0\0H\3\0\0\0\0\0\0\0
\10\0\0\0\0\0\0\0\3\0\0\0\4\0\0\0`$\36\0\0\0\0\0`$\36\0\0\0\0\0`$\36\0\0\0\0\0\34\0\0\0\0\0\
0\0\34\0\0\0\0\0\0\0\20\0\0\0\0\0\0\0\1\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\0\0\0\0\350t\2\0\0\0\0\0\0\350t\2\0\0\0\0\0\0\20\0\0\0\0\0\0\1\0\0\0\5\0\0\0\0\200\2\0\0\0\0\
0\0\200\2\0\0\0\0\0\0\200\2\0\0\0\0\0\0}A\26\0\0\0\0\0\0}A\26\0\0\0\0\0\0\0\20\0\0\0\0\0\0\0\1\0\0\
0\4\0\0\0\0\0\320\30\0\0\0\0\0\0\0\320\30\0\0\0\0\0\0\0\320\30\0\0\0\0\0\0\0\T\5\0\0\0\0\0\0\T\5\0\0\0\0\
0\0\20\0\0\0\0\0\0\0\1\0\0\0\6\0\0\0\0x+\36\0\0\0\0\0\0x;\36\0\0\0\0\0\0x;\36\0\0\0\0\0\0pK\0\0\0\0\0\
0      #\1\0\0\0\0\0\0\20\0\0\0\0\0\0\0\2\0\0\0\6\0\0\0      Y\36\0\0\0\0\0\0      i\36\0\0\0\0\0\0
i\36\0\0\0\0\0\0@\2\0\0\0\0\0\0\0@\2\0\0\0\0\0\0\0\10\0\0\0\0\0\0\0\0\4\0\0\0\4\0\0\0\210\3\0\0\0\0\
0\0\210\3\0\0\0\0\0\0\0\210\3\0\0\0\0\0\0\0      \0\0\0\0\0\0\0\0\
0\0\0\0\0\0\0\0\10\0\0\0\0\0\0\0\4\0\0\0\4\0\0\0\250\3\0\0\0\0\0\0\250\3\0\0\0\0\0\0\250\3\0\
0\0\0\0\0$0\0\0\0\0\0\0$0\0\0\0\0\0\0\0\4\0\0\0"... , 840, 64) = 840
```

```
mmap(NULL, 2055800, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7b3f7eba6000
```

```
mmap(0x7b3f7ebce000, 1462272, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7b3f7ebce000
```

```

    mmap(0x7b3f7ed33000, 352256, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x18d000) = 0x7b3f7ed33000

```

```

    mmap(0x7b3f7ed89000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3. 0x1e2000) = 0x7b3f7ed89000

```

```

    mmap(0x7b3f7ed8f000, 52856, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS,
-1, 0) = 0x7b3f7ed8f000

```

```

close(3) = 0
mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7b3f7eba3000
arch_prctl(ARCH_SET_FS, 0x7b3f7eba3740) = 0
set_tid_address(0x7b3f7eba3a10) = 5624
set_robust_list(0x7b3f7eba3a20, 24) = 0
rseq(0x7b3f7eba3680, 0x20, 0, 0x53053053) = 0
mprotect(0x7b3f7ed89000, 16384, PROT_READ) = 0
mprotect(0x403000, 4096, PROT_READ) = 0
mprotect(0x7b3f7edda000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
munmap(0x7b3f7ed9c000, 29003) = 0
getpid() = 5624
fstat(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}) = 0
getrandom("\x37\xfb\xd4\x70\x9d\x60\xa0\x47", 8, GRND_NONBLOCK) = 8
brk(NULL) = 0x2ef27000
brk(0x2ef48000) = 0x2ef48000
write(1, "Parent \320\277\321\200\320\276\321\206\320\265\321\201\321\201, PID:
5624\n", 33Parent nrocecc, PID: 5624
) = 33
pipe2([3, 4], 0) = 0
pipe2([5, 6], 0) = 0
pipe2([7, 8], 0) = 0
rt_sigprocmask(SIG_BLOCK, ~[], [], 8) = 0
clone(child_stack=NULL, flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLDstrace:
Process 5625 attached
, child_tidptr=0x7b3f7eba3a10) = 5625
[pid 5625] set_robust_list(0x7b3f7eba3a20, 24 <unfinished ...>
[pid 5624] rt_sigprocmask(SIG_SETMASK, [], <unfinished ...>
[pid 5625] <... set_robust_list resumed>) = 0
[pid 5624] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 5624] rt_sigprocmask(SIG_BLOCK, ~[], <unfinished ...>
[pid 5625] rt_sigprocmask(SIG_SETMASK, [], <unfinished ...>
[pid 5624] <... rt_sigprocmask resumed>[], 8) = 0
[pid 5625] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 5624] clone(child_stack=NULL,
flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLD <unfinished ...>
[pid 5625] close(4) = 0
[pid 5625] close(5strace: Process 5626 attached

```



```

)                                = 0
[pid 5626] set_robust_list(0x7b3f7eba3a20, 24 <unfinished ...>
[pid 5625] close(7 <unfinished ...>
[pid 5626] <... set_robust_list resumed>) = 0
[pid 5625] <... close resumed>          = 0
[pid 5624] <... clone resumed>, child_tidptr=0x7b3f7eba3a10) = 5626
[pid 5626] rt_sigprocmask(SIG_SETMASK, [], <unfinished ...>
[pid 5625] close(8 <unfinished ...>
[pid 5624] rt_sigprocmask(SIG_SETMASK, [], <unfinished ...>
[pid 5626] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 5624] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 5625] <... close resumed>          = 0
[pid 5624] close(3 <unfinished ...>
[pid 5626] close(3 <unfinished ...>
[pid 5625] dup2(3, 0 <unfinished ...>
[pid 5624] <... close resumed>          = 0
[pid 5626] <... close resumed>          = 0
[pid 5624] close(5 <unfinished ...>
[pid 5625] <... dup2 resumed>           = 0
[pid 5624] <... close resumed>          = 0
[pid 5626] close(4 <unfinished ...>
[pid 5624] close(6 <unfinished ...>
[pid 5625] dup2(6, 1 <unfinished ...>
[pid 5624] <... close resumed>          = 0
[pid 5626] <... close resumed>          = 0
[pid 5624] close(8 <unfinished ...>
[pid 5625] <... dup2 resumed>           = 1
[pid 5624] <... close resumed>          = 0
[pid 5626] close(6 <unfinished ...>
[pid 5624] write(1, "\320\222\320\262\320\265\320\264\320\270\321\202\320\265\321\201\321\202\321\200\320\276\320\272\321\203:\n", 29 <unfinished ...>

```

Введите строку:

```

[pid 5625] execve("./child1", ["child1"], 0x7ffeb13c4348 /* 27 vars */ <unfinished ...>
[pid 5624] <... write resumed>          = 29
[pid 5626] <... close resumed>          = 0
[pid 5624] fstat(0, <unfinished ...>
[pid 5626] close(7 <unfinished ...>

```

```

= 0
[pid 5624] <... fstat resumed>{st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...})
= 0
[pid 5626] <... close resumed> = 0
[pid 5624] read(0, <unfinished ...>
[pid 5626] dup2(5, 0) = 0
[pid 5626] dup2(8, 1) = 1
[pid 5626] execve("./child2", ["child2"], 0x7ffeb13c4348 /* 27 vars */) = 0
[pid 5625] <... execve resumed> = 0
[pid 5626] brk(NULL <unfinished ...>
[pid 5625] brk(NULL <unfinished ...>
[pid 5626] <... brk resumed> = 0x12a89000
[pid 5625] <... brk resumed> = 0x32058000
[pid 5626] mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0
<unfinished ...>
[pid 5625] mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0
<unfinished ...>
[pid 5626] <... mmap resumed> = 0x73a4a47ea000
[pid 5625] <... mmap resumed> = 0x7cebbff75000
[pid 5626] access("/etc/ld.so.preload", R_OK <unfinished ...>
[pid 5625] access("/etc/ld.so.preload", R_OK <unfinished ...>
[pid 5626] <... access resumed> = -1 ENOENT (No such file or directory)
[pid 5625] <... access resumed> = -1 ENOENT (No such file or directory)
[pid 5626] openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC <unfinished ...>
[pid 5625] openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC <unfinished ...>
[pid 5626] <... openat resumed> = 3
[pid 5626] fstat(3, <unfinished ...>
[pid 5625] <... openat resumed> = 4
[pid 5626] <... fstat resumed>{st_mode=S_IFREG|0644, st_size=29003, ...}) = 0
[pid 5625] fstat(4, <unfinished ...>
[pid 5626] mmap(NULL, 29003, PROT_READ, MAP_PRIVATE, 3, 0 <unfinished ...>
[pid 5625] <... fstat resumed>{st_mode=S_IFREG|0644, st_size=29003, ...}) = 0
[pid 5626] <... mmap resumed> = 0x73a4a47e2000
[pid 5625] mmap(NULL, 29003, PROT_READ, MAP_PRIVATE, 4, 0 <unfinished ...>
[pid 5626] close(3) = 0
[pid 5625] <... mmap resumed> = 0x7cebbff6d000
[pid 5626] openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC
<unfinished ...>
[pid 5625] close(4 <unfinished ...>
[pid 5626] <... openat resumed> = 3

```

```
[pid 5625] <... close resumed>)          = 0
```



```

[pid 5625] <... mmap resumed>) = 0x7cebbfd77000
[pid 5626] <... mmap resumed>) = 0x73a4a4779000
[pid 5625] mmap(0x7cebbfd9f000, 1462272, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 4, 0x28000 <unfinished ...>
[pid 5626] mmap(0x73a4a47cf000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1e2000 <unfinished ...>
[pid 5625] <... mmap resumed>) = 0x7cebbfd9f000
[pid 5626] <... mmap resumed>) = 0x73a4a47cf000
[pid 5625] mmap(0x7cebbff04000, 352256, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 4, 0x18d000 <unfinished ...>
[pid 5626] mmap(0x73a4a47d5000, 52856, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0 <unfinished ...>
[pid 5625] <... mmap resumed>) = 0x7cebbff04000
[pid 5626] <... mmap resumed>) = 0x73a4a47d5000
[pid 5625] mmap(0x7cebbff5a000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 4, 0x1e2000 <unfinished ...>
[pid 5626] close(3 <unfinished ...>
[pid 5625] <... mmap resumed>) = 0x7cebbff5a000
[pid 5626] <... close resumed>) = 0
[pid 5625] mmap(0x7cebbff60000, 52856, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0 <unfinished ...>
[pid 5626] mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0
<unfinished ...>
[pid 5625] <... mmap resumed>) = 0x7cebbff60000
[pid 5626] <... mmap resumed>) = 0x73a4a45e9000
[pid 5626] arch_prctl(ARCH_SET_FS, 0x73a4a45e9740 <unfinished ...>
[pid 5625] close(4 <unfinished ...>
[pid 5626] <... arch_prctl resumed>) = 0
[pid 5625] <... close resumed>) = 0
[pid 5626] set_tid_address(0x73a4a45e9a10 <unfinished ...>
[pid 5625] mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0
<unfinished ...>
[pid 5626] <... set_tid_address resumed>) = 5626
[pid 5625] <... mmap resumed>) = 0x7cebbfd74000
[pid 5626] set_robust_list(0x73a4a45e9a20, 24 <unfinished ...>
[pid 5625] arch_prctl(ARCH_SET_FS, 0x7cebbfd74740 <unfinished ...>
[pid 5626] <... set_robust_list resumed>) = 0
[pid 5625] <... arch_prctl resumed>) = 0
[pid 5626] rseq(0x73a4a45e9680, 0x20, 0, 0x53053053 <unfinished ...>
[pid 5625] set_tid_address(0x7cebbfd74a10 <unfinished ...>

```

```

[pid 5626] <... rseq resumed>) = 0
[pid 5625] <... set_tid_address resumed>) = 5625
[pid 5625] set_robust_list(0x7cebbfd74a20, 24 <unfinished ...>
[pid 5626] mprotect(0x73a4a47cf000, 16384, PROT_READ <unfinished ...>
[pid 5625] <... set_robust_list resumed>) = 0
[pid 5626] <... mprotect resumed>) = 0
[pid 5625] rseq(0x7cebbfd74680, 0x20, 0, 0x53053053 <unfinished ...>
[pid 5626] mprotect(0x403000, 4096, PROT_READ <unfinished ...>
[pid 5625] <... rseq resumed>) = 0
[pid 5626] <... mprotect resumed>) = 0
[pid 5626] mprotect(0x73a4a4820000, 8192, PROT_READ <unfinished ...>
[pid 5625] mprotect(0x7cebbff5a000, 16384, PROT_READ <unfinished ...>
[pid 5626] <... mprotect resumed>) = 0
[pid 5625] <... mprotect resumed>) = 0
[pid 5626] prlimit64(0, RLIMIT_STACK, NULL, <unfinished ...>
[pid 5625] mprotect(0x403000, 4096, PROT_READ <unfinished ...>
[pid 5626] <... prlimit64 resumed>{rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
[pid 5625] <... mprotect resumed>) = 0
[pid 5626] munmap(0x73a4a47e2000, 29003 <unfinished ...>
[pid 5625] mprotect(0x7cebbffab000, 8192, PROT_READ <unfinished ...>
[pid 5626] <... munmap resumed>) = 0
[pid 5625] <... mprotect resumed>) = 0
[pid 5626] getpid( <unfinished ...>
[pid 5625] prlimit64(0, RLIMIT_STACK, NULL, <unfinished ...>
[pid 5626] <... getpid resumed>) = 5626
[pid 5625] <... prlimit64 resumed>{rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
[pid 5626] fstat(1, <unfinished ...>
[pid 5625] munmap(0x7cebbff6d000, 29003 <unfinished ...>
[pid 5626] <... fstat resumed>{st_mode=S_IFIFO|0600, st_size=0, ...}) = 0
[pid 5625] <... munmap resumed>) = 0
[pid 5626] getrandom("\x66\x9d\xd5\x5b\x50\xac\xd9\x56", 8, GRND_NONBLOCK) = 8
[pid 5625] getpid( <unfinished ...>
[pid 5626] brk(NULL <unfinished ...>
[pid 5625] <... getpid resumed>) = 5625
[pid 5626] <... brk resumed>) = 0x12a89000
[pid 5626] brk(0x12aaa000 <unfinished ...>
[pid 5625] fstat(1, <unfinished ...>
[pid 5626] <... brk resumed>) = 0x12aaa000

```

```

[pid 5625] <... fstat resumed>{st_mode=S_IFIFO|0600, st_size=0, ...}) = 0
[pid 5626] read(0, <unfinished ...>
[pid 5625] getrandom("\xf9\xe9\x96\x3b\x58\x57\x28\xda", 8, GRND_NONBLOCK) = 8
[pid 5625] brk(NULL) = 0x32058000
[pid 5625] brk(0x32079000) = 0x32079000
[pid 5625] read(0, Hello World!
<unfinished ...>
[pid 5624] <... read resumed>"Hello World!\n", 1024) = 13
[pid 5624] write(4, "Hello World!\n", 13) = 13
[pid 5625] <... read resumed>"Hello World!\n", 1024) = 13
[pid 5624] read(7, <unfinished ...>
[pid 5625] write(1, "hello world!\n", 13 <unfinished ...>
[pid 5626] <... read resumed>"hello world!\n", 1024) = 13
[pid 5625] <... write resumed> = 13
[pid 5626] write(1, "hello_world!\n", 13 <unfinished ...>
[pid 5625] read(0, <unfinished ...>
[pid 5626] <... write resumed> = 13
[pid 5624] <... read resumed>"hello_world!\n", 1023) = 13
[pid 5624] write(1, "\320\230\321\202\320\276\320\263: hello_world!\n", 23 <unfinished
...>
[pid 5626] read(0, Итор: hello_world!
<unfinished ...>
[pid 5624] <... write resumed> = 23
[pid 5624] read(0, "", 1024) = 0
[pid 5624] close(4) = 0
[pid 5625] <... read resumed>"", 1024) = 0
[pid 5624] close(7) = 0
[pid 5625] write(1, "Child1 \320\267\320\260\320\277\321\203\321\211\320\265\320\275,
PID: 5625\nChild1 \320\267\320\260\320\262\320\265\321\200\321\210\320\265\320\275\n", 57
<unfinished ...>
[pid 5624] wait4(5625, <unfinished ...>
[pid 5626] <... read resumed>"Child1
\320\267\320\260\320\277\321\203\321\211\320\265\320\275, PID: 5625\nChild1
\320\267\320\260\320\262\320\265\321\200\321\210\320\265\320\275\n", 1024) = 57
[pid 5625] <... write resumed> = 57
[pid 5626] write(1,
"Child1_\320\267\320\260\320\277\321\203\321\211\320\265\320\275,_PID:_5625\nChild1_\320\267
\320\260\320\262\320\265\321\200\321\210\320\265\320\275\n", 57 <unfinished ...>
[pid 5625] exit_group(0 <unfinished ...>
[pid 5626] <... write resumed> = -1 EPIPE (Broken pipe)

```

```

[pid 5626] --- SIGPIPE {si_signo=SIGPIPE, si_code=SI_USER, si_pid=5626, si_uid=0} --
-
[pid 5625] <... exit_group resumed>    = ?
[pid 5626] +++ killed by SIGPIPE +++
[pid 5625] +++ exited with 0 +++
<... wait4 resumed>NULL, 0, NULL)      = 5625
--- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_KILLED, si_pid=5626, si_uid=0,
si_status=SIGPIPE, si_utime=0, si_stime=0} ---
wait4(5626, NULL, 0, NULL)            = 5626
write(1, "Parent \320\267\320\260\320\262\320\265\321\200\321\210\320\265\320\275.\n",
25Parent завершен.
) = 25
exit_group(0)                         = ?
+++ exited with 0 +++

```

## Вывод

В ходе выполнения лабораторной работы я научился создавать пайпы, процессы, понял, как программа взаимодействует с процессами, научился перенаправлять потоки ввода/вывода, то есть изучил инструменты, которые помогают работать напрямую с компьютером, что, по-моему, очень важно. Основная сложность была в правильном закрытии файловых дескрипторов – если они оставались открытыми, программа зависала. Лабораторная работа понравилась, теперь стало чуть понятнее, как процессы работают вместе, взаимодействуют, как передают друг другу информацию.