

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: Ле Шон Лыонг

Преподаватель: Миронов Е.С.

Оценка: \_\_\_\_\_

Дата: 06.11.25

Москва, 2025

# Постановка задачи

## Вариант 14.

Разработать программу-родитель, которая создает два дочерних процесса. Взаимодействие между процессами организуется через неименованные каналы (pipes). Первый дочерний процесс (child1) должен преобразовывать все символы входной строки в нижний регистр, а второй дочерний процесс (child2) должен удалять все подряд идущие пробелы, оставляя только одиночные пробелы. Родительский процесс читает строки из стандартного ввода, передает их через цепочку процессов и выводит окончательный результат.

## Общий метод и алгоритм решения

- `id_t fork(void);` — создает дочерний процесс
- `int pipe(int fd[2]);` — создает неименованный канал
- `ssize_t read(int fd, void *buf, size_t count);` — чтение из файлового дескриптора
- `ssize_t write(int fd, const void *buf, size_t count);` — запись в файловый дескриптор
- `int dup2(int oldfd, int newfd);` — перенаправление стандартных потоков
- `int execl(const char *path, const char *arg, ...);` — запуск исполняемого файла
- `pid_t waitpid(pid_t pid, int *status, int options);` — ожидание завершения процесса

Алгоритм работы программы:

1. **Инициализация каналов:** Родительский процесс создает три канала:
  - `pipe1`: родитель → `child1`
  - `pipe2`: `child1` → `child2`
  - `pipe3`: `child2` → родитель
2. **Создание процессов:** Родитель создает два дочерних процесса с помощью `fork()`
3. **Настройка дочерних процессов:**
  - `Child1`: перенаправляет `stdin` на `pipe1`, `stdout` на `pipe2`, выполняет программу `child1`
  - `Child2`: перенаправляет `stdin` на `pipe2`, `stdout` на `pipe3`, выполняет программу `child2`
4. **Обработка данных:**
  - Родитель читает строки из `stdin` и пишет в `pipe1`
  - `Child1` читает из `pipe1`, переводит в нижний регистр, пишет в `pipe2`
  - `Child2` читает из `pipe2`, удаляет лишние пробелы, пишет в `pipe3`
  - Родитель читает из `pipe3` и выводит результат
5. **Завершение:** При окончании ввода родитель закрывает каналы и ожидает завершения дочерних процессов

## Код программы

### parent.c

```
#define _GNU_SOURCE
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
```

```

#include <sys/wait.h>
#include <string.h>
#include <errno.h>

#define BUF 4096

ssize_t write_all(int fd, const void *buf, size_t count) {
    const char *p = buf;
    size_t left = count;
    while (left > 0) {
        ssize_t n = write(fd, p, left);
        if (n < 0) {
            if (errno == EINTR) continue;
            return -1;
        }
        left -= n;
        p += n;
    }
    return count;
}

ssize_t read_line(int fd, char *buf, size_t cap) {
    size_t pos = 0;
    while (pos + 1 < cap) {
        char c;
        ssize_t n = read(fd, &c, 1);
        if (n <= 0) break;
        buf[pos++] = c;
        if (c == '\n') break;
    }
    buf[pos] = '\0';
    return pos;
}

int main() {
    int p1[2], p2[2], p3[2];

    if (pipe(p1) == -1 || pipe(p2) == -1 || pipe(p3) == -1) {
        perror("pipe");
        exit(1);
    }

    pid_t c1 = fork();
    if (c1 == 0) {
        dup2(p1[0], STDIN_FILENO);
        dup2(p2[1], STDOUT_FILENO);

        close(p1[0]); close(p1[1]);
        close(p2[0]); close(p2[1]);
        close(p3[0]); close(p3[1]);

        execl("./child1", "child1", NULL);
        perror("exec child1");
        _exit(1);
    }

    pid_t c2 = fork();
    if (c2 == 0) {
        dup2(p2[0], STDIN_FILENO);
        dup2(p3[1], STDOUT_FILENO);

        close(p1[0]); close(p1[1]);
        close(p2[0]); close(p2[1]);
        close(p3[0]); close(p3[1]);
    }
}

```

```

        execl("./child2", "child2", NULL);
        perror("exec child2");
        _exit(1);
    }

    close(p1[0]);
    close(p2[0]); close(p2[1]);
    close(p3[1]);

    char line[BUF], result[BUF];
    printf("Введите строку:\n");

    while (fgets(line, sizeof(line), stdin)) {
        write_all(p1[1], line, strlen(line));

        ssize_t n = read_line(p3[0], result, sizeof(result));
        if (n <= 0) break;

        printf("Результат: %s", result);
    }

    close(p1[1]);
    close(p3[0]);
    waitpid(c1, NULL, 0);
    waitpid(c2, NULL, 0);

    return 0;
}

```

#### **child1.c**

```

#include <stdio.h>
#include <unistd.h>
#include <ctype.h>

int main() {
    char buf[4096];
    ssize_t n;

    while ((n = read(STDIN_FILENO, buf, sizeof(buf))) > 0) {
        for (ssize_t i = 0; i < n; i++) {
            buf[i] = tolower((unsigned char)buf[i]);
        }
        write(STDOUT_FILENO, buf, n);
    }
    return 0;
}

```

#### **child2.c**

```

#include <stdio.h>
#include <unistd.h>

int main() {
    char in[4096], out[4096];
    ssize_t n;
    int last_space = 0;

    while ((n = read(STDIN_FILENO, in, sizeof(in))) > 0) {
        ssize_t pos = 0;

        for (ssize_t i = 0; i < n; i++) {
            if (in[i] == ' ') {
                if (!last_space) {
                    out[pos++] = ' ';
                }
                last_space = 1;
            }
        }
    }
}

```

```

    } else {
        out[pos++] = in[i];
        last_space = 0;
    }
}
write(STDOUT_FILENO, out, pos);
}
return 0;
}

```

## Протокол работы программы

```

on@DESKTOP-F8Q4L97:/mnt/c/Users/Son/LabsOS$ ./parent
Введите строку:
HELLO WORLD! hELLO WoRLD!
hello world! hello world!

```

### Strace:

```

**execve("./parent", [ "./parent" ], 0x7ffd796dc3c8 /* 25 vars */) = 0**
brk(NULL) = 0x6407c586a000
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7b239c26b000
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=52259, ...}) = 0
mmap(NULL, 52259, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7b239c25e000
close(3) = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\13\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\220\243\2\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"..., 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"..., 784, 64) = 784
mmap(NULL, 2170256, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7b239c000000
mmap(0x7b239c028000, 1605632, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7b239c028000
mmap(0x7b239c1b0000, 323584, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x1b0000) = 0x7b239c1b0000
mmap(0x7b239c1ff000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1fe000) = 0x7b239c1ff000
mmap(0x7b239c205000, 52624, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7b239c205000
close(3) = 0
mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7b239c25b000
arch_prctl(ARCH_SET_FS, 0x7b239c25b740) = 0
set_tid_address(0x7b239c25ba10) = 436
set_robust_list(0x7b239c25ba20, 24) = 0
rseq(0x7b239c25c060, 0x20, 0, 0x53053053) = 0
mprotect(0x7b239c1ff000, 16384, PROT_READ) = 0
mprotect(0x6407b87e4000, 4096, PROT_READ) = 0
mprotect(0x7b239c2a3000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
munmap(0x7b239c25e000, 52259) = 0
pipe2([3, 4], 0) = 0
pipe2([5, 6], 0) = 0
pipe2([7, 8], 0) = 0
clone(child_stack=NULL,

```

```

flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLD,
child_tidptr=0x7b239c25ba10) = 437
437 set_robust_list(0x7b239c25ba20, 24 <unfinished ...>
436 clone(child_stack=NULL,
flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLD <unfinished ...>
437 <... set_robust_list resumed>) = 0
436 <... clone resumed>, child_tidptr=0x7b239c25ba10) = 438
438 set_robust_list(0x7b239c25ba20, 24 <unfinished ...>
436 close(3 <unfinished ...>
437 dup2(3, 0 <unfinished ...>
436 <... close resumed>) = 0
437 <... dup2 resumed>) = 0
436 close(5 <unfinished ...>
438 <... set_robust_list resumed>) = 0
436 <... close resumed>) = 0
437 dup2(6, 1 <unfinished ...>
436 close(6 <unfinished ...>
438 dup2(5, 0 <unfinished ...>
436 <... close resumed>) = 0
437 <... dup2 resumed>) = 1
436 close(8 <unfinished ...>
438 <... dup2 resumed>) = 0
436 <... close resumed>) = 0
437 close(3 <unfinished ...>
436 fstat(1, <unfinished ...>
438 dup2(8, 1 <unfinished ...>
436 <... fstat resumed>{st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}) = 0
437 <... close resumed>) = 0
436 getrandom( <unfinished ...>
438 <... dup2 resumed>) = 1
436 <... getrandom resumed>"\xbf\xa4\xc9\xe3\xd4\x75\xc2", 8, GRND_NONBLOCK) = 8
437 close(4 <unfinished ...>
436 brk(NULL <unfinished ...>
438 close(3 <unfinished ...>
436 <... brk resumed>) = 0x6407c586a000
437 <... close resumed>) = 0
436 brk(0x6407c588b000 <unfinished ...>
438 <... close resumed>) = 0
436 <... brk resumed>) = 0x6407c588b000
437 close(5 <unfinished ...>
436 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 ...>
438 close(4 <unfinished ...>
436 <... write resumed>) = 29
437 <... close resumed>) = 0
436 fstat(0, <unfinished ...>
438 <... close resumed>) = 0
436 <... fstat resumed>{st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}) = 0
437 close(6 <unfinished ...>
436 read(0, <unfinished ...>
438 close(5 <unfinished ...>
437 <... close resumed>) = 0
438 <... close resumed>) = 0
437 close(7 <unfinished ...>
438 close(6 <unfinished ...>

```

```

437 <... close resumed>) = 0
438 <... close resumed>) = 0
437 close(8 <unfinished ...>
438 close(7 <unfinished ...>
437 <... close resumed>) = 0
438 <... close resumed>) = 0
437 **execve("./child1", ["child1"], 0x7fff0d3f0a98 /* 25 vars / <unfinished ...>*)
438 close(8) = 0
438 **execve("./child2", ["child2"], 0x7fff0d3f0a98 /* 25 vars */) = 0**
438 brk(NULL <unfinished ...>
437 <... execve resumed>) = 0
438 <... brk resumed>) = 0x5b4ad6732000
437 brk(NULL <unfinished ...>
438 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0
<unfinished ...>
437 <... brk resumed>) = 0x56f2cacea000
438 <... mmap resumed>) = 0x7fc0814c4000
437 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0
<unfinished ...>
438 access("/etc/ld.so.preload", R_OK <unfinished ...>
437 <... mmap resumed>) = 0x704d8bdd4000
438 <... access resumed>) = -1 ENOENT (No such file or directory)
437 access("/etc/ld.so.preload", R_OK <unfinished ...>
438 openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC <unfinished ...>
437 <... access resumed>) = -1 ENOENT (No such file or directory)
438 <... openat resumed>) = 3
437 openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC <unfinished ...>
438 fstat(3, <unfinished ...>
437 <... openat resumed>) = 3
438 <... fstat resumed>{st_mode=S_IFREG|0644, st_size=52259, ...}) = 0
437 fstat(3, <unfinished ...>
438 mmap(NULL, 52259, PROT_READ, MAP_PRIVATE, 3, 0 <unfinished ...>
437 <... fstat resumed>{st_mode=S_IFREG|0644, st_size=52259, ...}) = 0
438 <... mmap resumed>) = 0x7fc0814b7000
437 mmap(NULL, 52259, PROT_READ, MAP_PRIVATE, 3, 0 <unfinished ...>
438 close(3 <unfinished ...>
437 <... mmap resumed>) = 0x704d8bdc7000
438 <... close resumed>) = 0
437 close(3 <unfinished ...>
438 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC <unfinished ...>
437 <... close resumed>) = 0
438 <... openat resumed>) = 3
437 openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC <unfinished ...>
438 read(3, <unfinished ...>
437 <... openat resumed>) = 3
438 <... read resumed>"\177ELF2\1\13\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
437 read(3, <unfinished ...>
438 pread64(3, <unfinished ...>
437 <... read resumed>"\177ELF2\1\13\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
438 <... 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\0\0\0"..., 784, 64) =
784
437 pread64(3, <unfinished ...>
438 fstat(3, <unfinished ...>
437 <... 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\0\0\0"..., 784, 64) =
784

```

438 <... fstat resumed>{st\_mode=S\_IFREG|0755, st\_size=2125328, ...}) = 0  
437 fstat(3, <unfinished ...>  
438 pread64(3, <unfinished ...>  
437 <... fstat resumed>{st\_mode=S\_IFREG|0755, st\_size=2125328, ...}) = 0  
438 <... 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  
437 pread64(3, <unfinished ...>  
438 mmap(NULL, 2170256, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0 <unfinished ...>  
437 <... 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  
438 <... mmap resumed>) = 0x7fc081200000  
437 mmap(NULL, 2170256, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0 <unfinished ...>  
438 mmap(0x7fc081228000, 1605632, PROT\_READ|PROT\_EXEC,  
MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000 <unfinished ...>  
437 <... mmap resumed>) = 0x704d8ba00000  
438 <... mmap resumed>) = 0x7fc081228000  
437 mmap(0x704d8ba28000, 1605632, PROT\_READ|PROT\_EXEC,  
MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000 <unfinished ...>  
438 mmap(0x7fc0813b0000, 323584, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE,  
3, 0x1b0000 <unfinished ...>  
437 <... mmap resumed>) = 0x704d8ba28000  
438 <... mmap resumed>) = 0x7fc0813b0000  
437 mmap(0x704d8bbb0000, 323584, PROT\_READ,  
MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1b0000 <unfinished ...>  
438 mmap(0x7fc0813ff000, 24576, PROT\_READ|PROT\_WRITE,  
MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1fe000 <unfinished ...>  
437 <... mmap resumed>) = 0x704d8bbb0000  
438 <... mmap resumed>) = 0x7fc0813ff000  
437 mmap(0x704d8bbff000, 24576, PROT\_READ|PROT\_WRITE,  
MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1fe000 <unfinished ...>  
438 mmap(0x7fc081405000, 52624, PROT\_READ|PROT\_WRITE,  
MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0 <unfinished ...>  
437 <... mmap resumed>) = 0x704d8bbff000  
438 <... mmap resumed>) = 0x7fc081405000  
437 mmap(0x704d8bc05000, 52624, PROT\_READ|PROT\_WRITE,  
MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0 <unfinished ...>  
438 close(3 <unfinished ...>  
437 <... mmap resumed>) = 0x704d8bc05000  
438 <... close resumed>) = 0  
437 close(3 <unfinished ...>  
438 mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0  
<unfinished ...>  
437 <... close resumed>) = 0  
438 <... mmap resumed>) = 0x7fc0814b4000  
437 mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0  
<unfinished ...>  
438 arch\_prctl(ARCH\_SET\_FS, 0x7fc0814b4740 <unfinished ...>  
437 <... mmap resumed>) = 0x704d8bdc4000  
438 <... arch\_prctl resumed>) = 0  
437 arch\_prctl(ARCH\_SET\_FS, 0x704d8bdc4740 <unfinished ...>  
438 set\_tid\_address(0x7fc0814b4a10 <unfinished ...>  
437 <... arch\_prctl resumed>) = 0  
438 <... set\_tid\_address resumed>) = 438  
437 set\_tid\_address(0x704d8bdc4a10 <unfinished ...>  
438 set\_robust\_list(0x7fc0814b4a20, 24 <unfinished ...>

```

437 <... set_tid_address resumed>) = 437
438 <... set_robust_list resumed>) = 0
437 set_robust_list(0x704d8bdc4a20, 24 <unfinished ...>
438 rseq(0x7fc0814b5060, 0x20, 0, 0x53053053 <unfinished ...>
437 <... set_robust_list resumed>) = 0
438 <... rseq resumed>) = 0
437 rseq(0x704d8bdc5060, 0x20, 0, 0x53053053 <unfinished ...>
438 mprotect(0x7fc0813ff000, 16384, PROT_READ <unfinished ...>
437 <... rseq resumed>) = 0
438 <... mprotect resumed>) = 0
437 mprotect(0x704d8bbff000, 16384, PROT_READ <unfinished ...>
438 mprotect(0x5b4ac3130000, 4096, PROT_READ <unfinished ...>
437 <... mprotect resumed>) = 0
438 <... mprotect resumed>) = 0
437 mprotect(0x56f292c42000, 4096, PROT_READ <unfinished ...>
438 mprotect(0x7fc0814fc000, 8192, PROT_READ <unfinished ...>
437 <... mprotect resumed>) = 0
438 <... mprotect resumed>) = 0
437 mprotect(0x704d8be0c000, 8192, PROT_READ <unfinished ...>
438 prlimit64(0, RLIMIT_STACK, NULL, <unfinished ...>
437 <... mprotect resumed>) = 0
438 <... prlimit64 resumed>{rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
437 prlimit64(0, RLIMIT_STACK, NULL, <unfinished ...>
438 munmap(0x7fc0814b7000, 52259 <unfinished ...>
437 <... prlimit64 resumed>{rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
438 <... munmap resumed>) = 0
437 munmap(0x704d8bdc7000, 52259 <unfinished ...>
438 read(0, <unfinished ...>
437 <... munmap resumed>) = 0
437 read(0, <unfinished ...>
436 <... read resumed>"HELLO WORLD! hELLO WoRLD!\n", 1024) = 28
436 write(4, "HELLO WORLD! hELLO WoRLD!\n", 28) = 28
437 <... read resumed>"HELLO WORLD! hELLO WoRLD!\n", 4096) = 28
436 read(7, <unfinished ...>
437 write(1, "hello world! hello world!\n", 28 <unfinished ...>
438 <... read resumed>"hello world! hello world!\n", 4096) = 28
437 <... write resumed>) = 28
438 write(1, "hello world! hello world!\n", 26 <unfinished ...>
437 read(0, <unfinished ...>
436 <... read resumed>"h", 1) = 1
438 <... write resumed>) = 26
436 read(7, <unfinished ...>
438 read(0, <unfinished ...>
436 <... read resumed>"e", 1) = 1
436 read(7, "l", 1) = 1
436 read(7, "l", 1) = 1
436 read(7, "o", 1) = 1
436 read(7, " ", 1) = 1
436 read(7, "w", 1) = 1
436 read(7, "o", 1) = 1
436 read(7, "r", 1) = 1
436 read(7, "l", 1) = 1
436 read(7, "d", 1) = 1
436 read(7, "!", 1) = 1
436 read(7, " ", 1) = 1

```

```

436 read(7, "h", 1) = 1
436 read(7, "e", 1) = 1
436 read(7, "l", 1) = 1
436 read(7, "l", 1) = 1
436 read(7, "o", 1) = 1
436 read(7, " ", 1) = 1
436 read(7, "w", 1) = 1
436 read(7, "o", 1) = 1
436 read(7, "r", 1) = 1
436 read(7, "l", 1) = 1
436 read(7, "d", 1) = 1
436 read(7, "!", 1) = 1
436 read(7, "\n", 1) = 1
436 write(1, "hello world! hello world!\n", 26) = 26
436 read(0, "", 1024) = 0
436 close(4) = 0
437 <... read resumed>"", 4096) = 0
436 close(7 <unfinished ...>
437 exit_group(0 <unfinished ...>
436 <... close resumed>) = 0
437 <... exit_group resumed>) = ?
436 wait4(437, <unfinished ...>
438 <... read resumed>"", 4096) = 0
437 +++ exited with 0 +++
436 <... wait4 resumed>NULL, 0, NULL) = 437
438 exit_group(0 <unfinished ...>
436 --- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=437, si_uid=1000, si_status=0,
si_utime=0, si_stime=0} ---
438 <... exit_group resumed>) = ?
436 wait4(438, <unfinished ...>
438 +++ exited with 0 +++
436 <... wait4 resumed>NULL, 0, NULL) = 438
436 --- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=438, si_uid=1000, si_status=0,
si_utime=0, si_stime=0} ---
436 exit_group(0) = ?
436 +++ exited with 0 +++

```

## Вывод

В ходе лабораторной работы была успешно реализована система взаимодействующих процессов через неименованные каналы. Программа корректно выполняет двухэтапную обработку текста: перевод в нижний регистр и удаление лишних пробелов. Основные трудности были связаны с правильным управлением файловыми дескрипторами и организацией передачи данных между процессами. Полученный опыт демонстрирует эффективность межпроцессного взаимодействия в UNIX-системах.