

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-210Б-23

Студент: Домкин П.П.

Преподаватель: Бахарев В.Д.

Оценка: \_\_\_\_\_

Дата: 10.10.24

Москва, 2024

# Постановка задачи

## Вариант 9.

В файле записаны команды вида: «число число число<newline>». Дочерний процесс производит деление первого числа команда, на последующие числа в команде, а результат выводит в стандартный поток вывода. Если происходит деление на 0, то тогда дочерний и родительский процесс завершают свою работу. Проверка деления на 0 должна осуществляться на стороне дочернего процесса. Числа имеют тип float. Количество чисел может быть произвольным.

## Общий метод и алгоритм решения

Использованные системные вызовы:

- `pid_t fork(void)`; – создает дочерний процесс.
- `int pipe(int *fd)`; – создаёт pipe и помещает дескрипторы в `fd[0]`, `fd[1]`, для чтения и записи.
- `int write(int fd, const void* buffer, int count)`; – записывает по дескриптору `fd` `count` байт из `buffer`.
- `int dup2(int fd1, int fd2)`; – перенаправляет дескрипторы.
- `int exec(char* path, const char* argc)`; – заменяет текущий процесс на процесс `path`, с аргументами `argc`;
- `int close(int fd)`; – закрывает дескриптор `fd`.
- `pid_t wait(int status)` — функция, которая приостанавливает выполнение текущего процесса до тех пор, пока дочерний процесс не завершится.
- `void exit(int number)`; – вызывает нормальное завершение программы с кодом `number`.

В рамках лабораторной работы я реализовал две программы на C: `Parent.c` и `Child.c`, которые работают совместно для выполнения вычислений на основе чисел из заданного файла. Программа `Parent.c` запрашивает у пользователя путь к файлу, создает pipe для связи с дочерним процессом и использует `fork()` для его создания. Дочерний процесс перенаправляет стандартный ввод на открытый файл и выполняет программу `Child.c`, а родительский процесс перенаправляет стандартный вывод в pipe. `Child.c` считывает числа, обрабатывает их, выполняя деление, и выводит результаты через pipe. В случае ошибок, таких как неверный ввод или деление на ноль, программа выводит соответствующие сообщения об ошибках. Также родительский процесс ожидает завершение дочернего процесса после вывода результатов.

## Код программы

### Parent.c

```
#include <unistd.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <stdlib.h>
#include <fcntl.h>
#include <string.h>
#include <errno.h>

#define BUFFER_SIZE 1024
```

```

int main() {
    int pipe1[2];
    pid_t child_pid;
    char buffer;
    char filename[BUFFER_SIZE];
    ssize_t bytesRead;

    write(STDOUT_FILENO, "Enter filename path: ", 22);
    bytesRead = read(STDIN_FILENO, filename, sizeof(filename));
    if (bytesRead <= 0) {
        write(STDERR_FILENO, "Error reading the file name\n", 29);
        exit(EXIT_FAILURE);
    }

    if (filename[bytesRead - 1] == '\n') {
        filename[bytesRead - 1] = '\0';
    }

    if (pipe(pipe1) == -1) {
        write(STDERR_FILENO, "Pipe creation error\n", 21);
        exit(EXIT_FAILURE);
    }

    child_pid = fork();
    if (child_pid == -1) {
        write(STDERR_FILENO, "Process creation error\n", 24);
        exit(EXIT_FAILURE);
    }

    if (child_pid == 0) {
        close(pipe1[0]);

        dup2(pipe1[1], STDOUT_FILENO);
        close(pipe1[1]);

        execl("./child", "", filename, NULL);
        write(STDERR_FILENO, "Error starting a child process\n", 32);
        exit(EXIT_FAILURE);
    } else {
        close(pipe1[1]);

        // Read data from pipe and print
        while (read(pipe1[0], &buffer, 1) > 0) {
            write(STDOUT_FILENO, &buffer, 1);
        }
        close(pipe1[0]);

        wait(NULL); // Waiting until finish child process
        exit(EXIT_SUCCESS);
    }

    return 0;
}

```

## Child.c

```
#include <errno.h>
#include <fcntl.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>

#define BUFFER_SIZE 1024

void HandleError(const char *message) {
    write(STDERR_FILENO, message, strlen(message));
    exit(EXIT_FAILURE);
}

int main(int argc, char *argv[]) {
    if (argc < 2) {
        HandleError("Error: need to specify the path to the file.\n");
    }

    int file = open(argv[1], O_RDONLY);
    if (file == -1) {
        HandleError("Error opening file.\n");
    }

    // Redefining the standard input to the file
    dup2(file, STDIN_FILENO);
    close(file); // Close original descriptor

    char buffer[BUFFER_SIZE];
    ssize_t bytesRead;
    float num_first, num_next;
    char *current;
    char output[BUFFER_SIZE];
    int output_len;
    int flag = 0;
    int divisionByZeroError = 0;
```

```
// Read strings
while ((bytesRead = read(STDIN_FILENO, buffer, BUFFER_SIZE)) > 0) {
    current = buffer;

    // Read first number
    while (current < buffer + bytesRead) {
        while (*current == ' ' || *current == '\t') {
            current++;
        }
        if (*current == '\n') {
            current++;
            continue;
        }

        if (current >= buffer + bytesRead) {
            break;
        }

        char *endptr;
        num_first = strtoc(current, &endptr);

        if (endptr == current) {
            output_len = snprintf(output, BUFFER_SIZE, "Error: Non-numeric value encountered.\n");
            write(STDOUT_FILENO, output, output_len);
            while (*current != '\n' && current < buffer + bytesRead) {
                current++;
            }
            if (*current == '\n') {
                current++;
            }
            continue;
        }

        current = endptr;
```

```

// Read other numbers
while (1) {
    while (*current == ' ' || *current == '\t') {
        current++;
    }

    if (*current == '\n' || current >= buffer + bytesRead) {
        break;
    }

    num_next = strtod(current, &endptr);
    flag = 0;
    if (endptr == current) {
        output_len = snprintf(output, BUFFER_SIZE, "Error: Non-numeric value encountered.\n");
        write(STDOUT_FILENO, output, output_len);
        while (*current != '\n' && current < buffer + bytesRead) {
            current++;
        }
        if (*current == '\n') {
            flag = 1;
            continue;
        }
    }

    if (num_next == 0.0 || current == NULL || current == buffer) {
        output_len = snprintf(output, BUFFER_SIZE, "Division by zero.\n");
        write(STDOUT_FILENO, output, output_len);
        exit(EXIT_FAILURE);
    }

    num_first /= num_next;
    current = endptr;
}

```

```

}
if (flag == 0) {
    output_len = snprintf(output, BUFFER_SIZE, "Result: %.6f\n", num_first); // Get answer string
    write(STDOUT_FILENO, output, output_len);
}else{
    NULL;
}
while (*current != '\n' && current < buffer + bytesRead) {
    current++;
}

if (*current == '\n') {
    current++;
}
}
}

if (bytesRead == -1) {
    HandleError("Error reading file.\n");
}

exit(EXIT_SUCCESS);
}

```

# Протокол работы программы

## Тестирование:

```
1 5.0 4
2
3 1
4 10 2 5 6 8 2
5 1 1 1
6 2 b c
7 1 2 1
8 1 0 0
9 125 5 5
10 125 5 0
11 125 0 5
12
```

```
pablo@Ardor-Pavel:~/Main/3,4 SEM/Lab0s1/Lab0S1/src$ ./parent
Enter filename path: ../file/test.txt
Result: 1.250000
Result: 1.000000
Result: 0.010417
Result: 1.000000
Error: Non-numeric value encountered.
Result: 0.500000
Division by zero.
pablo@Ardor-Pavel:~/Main/3,4 SEM/Lab0s1/Lab0S1/src$
```

```
3,4 SEM / Lab0S1 / Lab0S1 / file / = test.txt
1 5.0 4 1
2 10 2 5 6 8 2
3 2 7
4 2 b c
5 1 2 1
6 125 5 5
7 125 5 0
8 125 CHECK 5
9
```

```
Division by zero.
pablo@Ardor-Pavel:~/Main/3,4 SEM/Lab0s1/Lab0S1/src$ ./parent
Enter filename path: ../file/test.txt
Result: 1.250000
Result: 0.010417
Result: 0.285714
Error: Non-numeric value encountered.
Result: 0.500000
Result: 5.000000
Division by zero.
pablo@Ardor-Pavel:~/Main/3,4 SEM/Lab0s1/Lab0S1/src$
```

## Strace:

```
$ strace -f ./parent
```

```
execve("./parent", ["/parent"], 0x7ffe569caf38 /* 36 vars */) = 0
```

```
brk(NULL) = 0x55a8bdeb0000
```

```
arch_prctl(0x3001 /* ARCH_??? */, 0x7ffe166ddc90) = -1 EINVAL (Invalid argument)
```

```
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fdab0df8000
```

```
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
```

```
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=18823, ...}, AT_EMPTY_PATH) = 0
```

```
mmap(NULL, 18823, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7fdab0df3000
```

```
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\3\0>\0\1\0\0\0P\237\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

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

pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0I\17\357\204\3\$\f\221\2039x\324\224\323\236S"..., 68, 896) = 68

newfstatat(3, "", {st\_mode=S\_IFREG|0755, st\_size=2220400, ...}, AT\_EMPTY\_PATH) = 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, 2264656, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7fdab0bca000

mprotect(0x7fdab0bf2000, 2023424, PROT\_NONE) = 0

mmap(0x7fdab0bf2000, 1658880, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000) = 0x7fdab0bf2000

mmap(0x7fdab0d87000, 360448, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1bd000) = 0x7fdab0d87000

mmap(0x7fdab0de0000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x215000) = 0x7fdab0de0000

mmap(0x7fdab0de6000, 52816, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7fdab0de6000

**close(3) = 0**

mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7fdab0bc7000

arch\_prctl(ARCH\_SET\_FS, 0x7fdab0bc7740) = 0

set\_tid\_address(0x7fdab0bc7a10) = 125703

set\_robust\_list(0x7fdab0bc7a20, 24) = 0

rseq(0x7fdab0bc80e0, 0x20, 0, 0x53053053) = 0

mprotect(0x7fdab0de0000, 16384, PROT\_READ) = 0

mprotect(0x55a8bcf54000, 4096, PROT\_READ) = 0

mprotect(0x7fdab0e32000, 8192, PROT\_READ) = 0

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

`munmap(0x7fdab0df3000, 18823) = 0`

`write(1, "Enter filename path: \0", 22Enter filename path: ) = 22`

`read(0, ../file/test.txt`

`"../file/test.txt\n", 1024) = 17`

`pipe2([3, 4], 0) = 0`

`clone(child_stack=NULL,  
flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLDstrace:  
Process 125889 attached`

`, child_tidptr=0x7fdab0bc7a10) = 125889`

`[pid 125703] close(4 <unfinished ...>`

`[pid 125889] set_robust_list(0x7fdab0bc7a20, 24 <unfinished ...>`

`[pid 125703] <... close resumed>) = 0`

`[pid 125889] <... set_robust_list resumed>) = 0`

`[pid 125703] read(3, <unfinished ...>`

`[pid 125889] close(3) = 0`

`[pid 125889] dup2(4, 1) = 1`

`[pid 125889] close(4) = 0`

`[pid 125889] execve("./child", [""], "../file/test.txt"), 0x7ffe166dde68 /* 36 vars */) =  
0`

`[pid 125889] brk(NULL) = 0x55b56fbf5000`

`[pid 125889] arch_prctl(0x3001 /* ARCH_??? */, 0x7ffc19ded430) = -1 EINVAL  
(Invalid argument)`

`[pid 125889] mmap(NULL, 8192, PROT_READ|PROT_WRITE,  
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fc198b3d000`

`[pid 125889] access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or  
directory)`

`[pid 125889] openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC)  
= 3`

`[pid 125889] newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=18823, ...},  
AT_EMPTY_PATH) = 0`

`[pid 125889] mmap(NULL, 18823, PROT_READ, MAP_PRIVATE, 3, 0) =  
0x7fc198b38000`

`[pid 125889] close(3) = 0`



```

[pid 125889] openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6",
O_RDONLY|O_CLOEXEC) = 3

[pid 125889] read(3,
"\177ELF2\1\1\3\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\237\2\0\0\0\0\0"..., 832) = 832

[pid 125889] 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

[pid 125889] 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\0"..., 48, 848) = 48

[pid 125889] pread64(3,
"\4\0\0\0\24\0\0\0\3\0\0\0GNU\0I\17\357\204\3$\f\221\2039x\324\224\323\236S"..., 68,
896) = 68

[pid 125889] newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=2220400, ...},
AT_EMPTY_PATH) = 0

[pid 125889] 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

[pid 125889] mmap(NULL, 2264656, PROT_READ,
MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7fc19890f000

[pid 125889] mprotect(0x7fc198937000, 2023424, PROT_NONE) = 0

[pid 125889] mmap(0x7fc198937000, 1658880, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7fc198937000

[pid 125889] mmap(0x7fc198acc000, 360448, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1bd000) = 0x7fc198acc000

[pid 125889] mmap(0x7fc198b25000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x215000) = 0x7fc198b25000

[pid 125889] mmap(0x7fc198b2b000, 52816, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7fc198b2b000

[pid 125889] close(3) = 0

[pid 125889] mmap(NULL, 12288, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fc19890c000

[pid 125889] arch_prctl(ARCH_SET_FS, 0x7fc19890c740) = 0

[pid 125889] set_tid_address(0x7fc19890ca10) = 125889

[pid 125889] set_robust_list(0x7fc19890ca20, 24) = 0

[pid 125889] rseq(0x7fc19890d0e0, 0x20, 0, 0x53053053) = 0

[pid 125889] mprotect(0x7fc198b25000, 16384, PROT_READ) = 0

[pid 125889] mprotect(0x55b56fae4000, 4096, PROT_READ) = 0

```

```
[pid 125889] mprotect(0x7fc198b77000, 8192, PROT_READ) = 0
[pid 125889] prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024,
rlim_max=RLIM64_INFINITY}) = 0
[pid 125889] munmap(0x7fc198b38000, 18823) = 0
[pid 125889] openat(AT_FDCWD, "../file/test.txt", O_RDONLY) = 3
[pid 125889] dup2(3, 0) = 0
[pid 125889] close(3) = 0
[pid 125889] read(0, "5.0 4", 1024) = 124
[pid 125889] write(1, "Result: 1.250000\n", 17 <unfinished ...>
[pid 125703] <... read resumed>"R", 1) = 1
[pid 125889] <... write resumed> = 17
[pid 125703] write(1, "R", 1 <unfinished ...>
R[pid 125889] write(1, "Result: 0.010417\n", 17 <unfinished ...>
[pid 125703] <... write resumed> = 1
[pid 125889] <... write resumed> = 17
[pid 125703] read(3, <unfinished ...>
[pid 125889] write(1, "Result: 0.285714\n", 17 <unfinished ...>
[pid 125703] <... read resumed>"e", 1) = 1
[pid 125889] <... write resumed> = 17
[pid 125703] write(1, "e", 1 <unfinished ...>
[pid 125889] write(1, "Error: Non-numeric value encount...", 38e <unfinished ...>
[pid 125703] <... write resumed> = 1
[pid 125889] <... write resumed> = 38
[pid 125703] read(3, <unfinished ...>
[pid 125889] write(1, "Result: 0.500000\n", 17 <unfinished ...>
[pid 125703] <... read resumed>"s", 1) = 1
[pid 125889] <... write resumed> = 17
[pid 125703] write(1, "s", 1 <unfinished ...>
s[pid 125889] write(1, "Result: 5.000000\n", 17 <unfinished ...>
[pid 125703] <... write resumed> = 1
```

[pid 125889] <... write resumed>) = 17

[pid 125703] read(3, "u", 1) = 1

[pid 125889] write(1, "Division by zero.\n", 18 <unfinished ...>

[pid 125703] write(1, "u", 1 <unfinished ...>

u[pid 125889] <... write resumed>) = 18

[pid 125703] <... write resumed>) = 1

[pid 125889] exit\_group(1 <unfinished ...>

[pid 125703] read(3, <unfinished ...>

[pid 125889] <... exit\_group resumed>) = ?

[pid 125703] <... read resumed>"l", 1) = 1

[pid 125703] write(1, "l", 1 <unfinished ...>

[pid 125889] +++ exited with 1 +++

<... write resumed>) = ? ERESTARTSYS (To be restarted if  
SA\_RESTART is set)

--- SIGCHLD {si\_signo=SIGCHLD, si\_code=CLD\_EXITED, si\_pid=125889,  
si\_uid=1000, si\_status=1, si\_etime=0, si\_stime=1} ---

**write(1, "l", 1l) = 1**

**read(3, "t", 1) = 1**

**write(1, "t", 1t) = 1**

**read(3, ":", 1) = 1**

**write(1, ":", 1:) = 1**

**read(3, " ", 1) = 1**

**write(1, " ", 1) = 1**

**read(3, "1", 1) = 1**

**write(1, "1", 1l) = 1**

**read(3, ".", 1) = 1**

**write(1, ".", 1.) = 1**

**read(3, "2", 1) = 1**

**write(1, "2", 12) = 1**

**read(3, "5", 1) = 1**

<b>write(1, "5", 15)</b>	<b>= 1</b>
<b>read(3, "0", 1)</b>	<b>= 1</b>
<b>write(1, "0", 10)</b>	<b>= 1</b>
<b>read(3, "0", 1)</b>	<b>= 1</b>
<b>write(1, "0", 10)</b>	<b>= 1</b>
<b>read(3, "0", 1)</b>	<b>= 1</b>
<b>write(1, "0", 10)</b>	<b>= 1</b>
<b>read(3, "0", 1)</b>	<b>= 1</b>
<b>write(1, "0", 10)</b>	<b>= 1</b>
<b>read(3, "\n", 1)</b>	<b>= 1</b>
<b>write(1, "\n", 1</b>	
<b>)</b>	<b>= 1</b>
<b>read(3, "R", 1)</b>	<b>= 1</b>
<b>write(1, "R", 1R)</b>	<b>= 1</b>
<b>read(3, "e", 1)</b>	<b>= 1</b>
<b>write(1, "e", 1e)</b>	<b>= 1</b>
<b>read(3, "s", 1)</b>	<b>= 1</b>
<b>write(1, "s", 1s)</b>	<b>= 1</b>
<b>read(3, "u", 1)</b>	<b>= 1</b>
<b>write(1, "u", 1u)</b>	<b>= 1</b>
<b>read(3, "l", 1)</b>	<b>= 1</b>
<b>write(1, "l", 1l)</b>	<b>= 1</b>
<b>read(3, "t", 1)</b>	<b>= 1</b>
<b>write(1, "t", 1t)</b>	<b>= 1</b>
<b>read(3, ":", 1)</b>	<b>= 1</b>
<b>write(1, ":", 1:)</b>	<b>= 1</b>
<b>read(3, " ", 1)</b>	<b>= 1</b>
<b>write(1, " ", 1 )</b>	<b>= 1</b>
<b>read(3, "0", 1)</b>	<b>= 1</b>

<b>write(1, "0", 10)</b>	<b>= 1</b>
<b>read(3, ".", 1)</b>	<b>= 1</b>
<b>write(1, ".", 1.)</b>	<b>= 1</b>
<b>read(3, "0", 1)</b>	<b>= 1</b>
<b>write(1, "0", 10)</b>	<b>= 1</b>
<b>read(3, "1", 1)</b>	<b>= 1</b>
<b>write(1, "1", 11)</b>	<b>= 1</b>
<b>read(3, "0", 1)</b>	<b>= 1</b>
<b>write(1, "0", 10)</b>	<b>= 1</b>
<b>read(3, "4", 1)</b>	<b>= 1</b>
<b>write(1, "4", 14)</b>	<b>= 1</b>
<b>read(3, "1", 1)</b>	<b>= 1</b>
<b>write(1, "1", 11)</b>	<b>= 1</b>
<b>read(3, "7", 1)</b>	<b>= 1</b>
<b>write(1, "7", 17)</b>	<b>= 1</b>
<b>read(3, "\n", 1)</b>	<b>= 1</b>
<b>write(1, "\n", 1</b>	
<b>)</b>	<b>= 1</b>
<b>read(3, "R", 1)</b>	<b>= 1</b>
<b>write(1, "R", 1R)</b>	<b>= 1</b>
<b>read(3, "e", 1)</b>	<b>= 1</b>
<b>write(1, "e", 1e)</b>	<b>= 1</b>
<b>read(3, "s", 1)</b>	<b>= 1</b>
<b>write(1, "s", 1s)</b>	<b>= 1</b>
<b>read(3, "u", 1)</b>	<b>= 1</b>
<b>write(1, "u", 1u)</b>	<b>= 1</b>
<b>read(3, "l", 1)</b>	<b>= 1</b>
<b>write(1, "l", 1l)</b>	<b>= 1</b>
<b>read(3, "t", 1)</b>	<b>= 1</b>

<b>write(1, "t", 1t)</b>	<b>= 1</b>
<b>read(3, ":", 1)</b>	<b>= 1</b>
<b>write(1, ":", 1:)</b>	<b>= 1</b>
<b>read(3, " ", 1)</b>	<b>= 1</b>
<b>write(1, " ", 1 )</b>	<b>= 1</b>
<b>read(3, "0", 1)</b>	<b>= 1</b>
<b>write(1, "0", 10)</b>	<b>= 1</b>
<b>read(3, ".", 1)</b>	<b>= 1</b>
<b>write(1, ".", 1.)</b>	<b>= 1</b>
<b>read(3, "2", 1)</b>	<b>= 1</b>
<b>write(1, "2", 12)</b>	<b>= 1</b>
<b>read(3, "8", 1)</b>	<b>= 1</b>
<b>write(1, "8", 18)</b>	<b>= 1</b>
<b>read(3, "5", 1)</b>	<b>= 1</b>
<b>write(1, "5", 15)</b>	<b>= 1</b>
<b>read(3, "7", 1)</b>	<b>= 1</b>
<b>write(1, "7", 17)</b>	<b>= 1</b>
<b>read(3, "1", 1)</b>	<b>= 1</b>
<b>write(1, "1", 11)</b>	<b>= 1</b>
<b>read(3, "4", 1)</b>	<b>= 1</b>
<b>write(1, "4", 14)</b>	<b>= 1</b>
<b>read(3, "\n", 1)</b>	<b>= 1</b>
<b>write(1, "\n", 1</b>	
<b>)</b>	<b>= 1</b>
<b>read(3, "E", 1)</b>	<b>= 1</b>
<b>write(1, "E", 1E)</b>	<b>= 1</b>
<b>read(3, "r", 1)</b>	<b>= 1</b>
<b>write(1, "r", 1r)</b>	<b>= 1</b>
<b>read(3, "r", 1)</b>	<b>= 1</b>

<b>write(1, "r", 1r)</b>	<b>= 1</b>
<b>read(3, "o", 1)</b>	<b>= 1</b>
<b>write(1, "o", 1o)</b>	<b>= 1</b>
<b>read(3, "r", 1)</b>	<b>= 1</b>
<b>write(1, "r", 1r)</b>	<b>= 1</b>
<b>read(3, ":", 1)</b>	<b>= 1</b>
<b>write(1, ":", 1:)</b>	<b>= 1</b>
<b>read(3, " ", 1)</b>	<b>= 1</b>
<b>write(1, " ", 1 )</b>	<b>= 1</b>
<b>read(3, "N", 1)</b>	<b>= 1</b>
<b>write(1, "N", 1N)</b>	<b>= 1</b>
<b>read(3, "o", 1)</b>	<b>= 1</b>
<b>write(1, "o", 1o)</b>	<b>= 1</b>
<b>read(3, "n", 1)</b>	<b>= 1</b>
<b>write(1, "n", 1n)</b>	<b>= 1</b>
<b>read(3, "-", 1)</b>	<b>= 1</b>
<b>write(1, "-", 1-)</b>	<b>= 1</b>
<b>read(3, "n", 1)</b>	<b>= 1</b>
<b>write(1, "n", 1n)</b>	<b>= 1</b>
<b>read(3, "u", 1)</b>	<b>= 1</b>
<b>write(1, "u", 1u)</b>	<b>= 1</b>
<b>read(3, "m", 1)</b>	<b>= 1</b>
<b>write(1, "m", 1m)</b>	<b>= 1</b>
<b>read(3, "e", 1)</b>	<b>= 1</b>
<b>write(1, "e", 1e)</b>	<b>= 1</b>
<b>read(3, "r", 1)</b>	<b>= 1</b>
<b>write(1, "r", 1r)</b>	<b>= 1</b>
<b>read(3, "i", 1)</b>	<b>= 1</b>
<b>write(1, "i", 1i)</b>	<b>= 1</b>

<b>read(3, "c", 1)</b>	<b>= 1</b>
<b>write(1, "c", 1c)</b>	<b>= 1</b>
<b>read(3, " ", 1)</b>	<b>= 1</b>
<b>write(1, " ", 1 )</b>	<b>= 1</b>
<b>read(3, "v", 1)</b>	<b>= 1</b>
<b>write(1, "v", 1v)</b>	<b>= 1</b>
<b>read(3, "a", 1)</b>	<b>= 1</b>
<b>write(1, "a", 1a)</b>	<b>= 1</b>
<b>read(3, "l", 1)</b>	<b>= 1</b>
<b>write(1, "l", 1l)</b>	<b>= 1</b>
<b>read(3, "u", 1)</b>	<b>= 1</b>
<b>write(1, "u", 1u)</b>	<b>= 1</b>
<b>read(3, "e", 1)</b>	<b>= 1</b>
<b>write(1, "e", 1e)</b>	<b>= 1</b>
<b>read(3, " ", 1)</b>	<b>= 1</b>
<b>write(1, " ", 1 )</b>	<b>= 1</b>
<b>read(3, "e", 1)</b>	<b>= 1</b>
<b>write(1, "e", 1e)</b>	<b>= 1</b>
<b>read(3, "n", 1)</b>	<b>= 1</b>
<b>write(1, "n", 1n)</b>	<b>= 1</b>
<b>read(3, "c", 1)</b>	<b>= 1</b>
<b>write(1, "c", 1c)</b>	<b>= 1</b>
<b>read(3, "o", 1)</b>	<b>= 1</b>
<b>write(1, "o", 1o)</b>	<b>= 1</b>
<b>read(3, "u", 1)</b>	<b>= 1</b>
<b>write(1, "u", 1u)</b>	<b>= 1</b>
<b>read(3, "n", 1)</b>	<b>= 1</b>
<b>write(1, "n", 1n)</b>	<b>= 1</b>
<b>read(3, "t", 1)</b>	<b>= 1</b>



<code>write(1, "t", 1t)</code>	<code>= 1</code>
<code>read(3, "e", 1)</code>	<code>= 1</code>
<code>write(1, "e", 1e)</code>	<code>= 1</code>
<code>read(3, "r", 1)</code>	<code>= 1</code>
<code>write(1, "r", 1r)</code>	<code>= 1</code>
<code>read(3, "e", 1)</code>	<code>= 1</code>
<code>write(1, "e", 1e)</code>	<code>= 1</code>
<code>read(3, "d", 1)</code>	<code>= 1</code>
<code>write(1, "d", 1d)</code>	<code>= 1</code>
<code>read(3, ".", 1)</code>	<code>= 1</code>
<code>write(1, ".", 1.)</code>	<code>= 1</code>
<code>read(3, "\n", 1)</code>	<code>= 1</code>
<code>write(1, "\n", 1</code> <code>)</code>	<code>= 1</code>
<code>read(3, "R", 1)</code>	<code>= 1</code>
<code>write(1, "R", 1R)</code>	<code>= 1</code>
<code>read(3, "e", 1)</code>	<code>= 1</code>
<code>write(1, "e", 1e)</code>	<code>= 1</code>
<code>read(3, "s", 1)</code>	<code>= 1</code>
<code>write(1, "s", 1s)</code>	<code>= 1</code>
<code>read(3, "u", 1)</code>	<code>= 1</code>
<code>write(1, "u", 1u)</code>	<code>= 1</code>
<code>read(3, "l", 1)</code>	<code>= 1</code>
<code>write(1, "l", 1l)</code>	<code>= 1</code>
<code>read(3, "t", 1)</code>	<code>= 1</code>
<code>write(1, "t", 1t)</code>	<code>= 1</code>
<code>read(3, ":", 1)</code>	<code>= 1</code>
<code>write(1, ":", 1:)</code>	<code>= 1</code>
<code>read(3, " ", 1)</code>	<code>= 1</code>

<code>write(1, " ", 1)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, ".", 1)</code>	<code>= 1</code>
<code>write(1, ".", 1)</code>	<code>= 1</code>
<code>read(3, "5", 1)</code>	<code>= 1</code>
<code>write(1, "5", 15)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "\n", 1)</code>	<code>= 1</code>
<code>write(1, "\n", 1</code>	
<code>)</code>	<code>= 1</code>
<code>read(3, "R", 1)</code>	<code>= 1</code>
<code>write(1, "R", 1R)</code>	<code>= 1</code>
<code>read(3, "e", 1)</code>	<code>= 1</code>
<code>write(1, "e", 1e)</code>	<code>= 1</code>
<code>read(3, "s", 1)</code>	<code>= 1</code>
<code>write(1, "s", 1s)</code>	<code>= 1</code>
<code>read(3, "u", 1)</code>	<code>= 1</code>
<code>write(1, "u", 1u)</code>	<code>= 1</code>
<code>read(3, "l", 1)</code>	<code>= 1</code>

<code>write(1, "l", 1l)</code>	<code>= 1</code>
<code>read(3, "t", 1)</code>	<code>= 1</code>
<code>write(1, "t", 1t)</code>	<code>= 1</code>
<code>read(3, ":", 1)</code>	<code>= 1</code>
<code>write(1, ":", 1:)</code>	<code>= 1</code>
<code>read(3, " ", 1)</code>	<code>= 1</code>
<code>write(1, " ", 1 )</code>	<code>= 1</code>
<code>read(3, "5", 1)</code>	<code>= 1</code>
<code>write(1, "5", 15)</code>	<code>= 1</code>
<code>read(3, ".", 1)</code>	<code>= 1</code>
<code>write(1, ".", 1.)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "0", 1)</code>	<code>= 1</code>
<code>write(1, "0", 10)</code>	<code>= 1</code>
<code>read(3, "\n", 1)</code>	<code>= 1</code>
<code>write(1, "\n", 1</code>	
<code>)</code>	<code>= 1</code>
<code>read(3, "D", 1)</code>	<code>= 1</code>
<code>write(1, "D", 1D)</code>	<code>= 1</code>
<code>read(3, "i", 1)</code>	<code>= 1</code>

<b>write(1, "i", 1i)</b>	<b>= 1</b>
<b>read(3, "v", 1)</b>	<b>= 1</b>
<b>write(1, "v", 1v)</b>	<b>= 1</b>
<b>read(3, "i", 1)</b>	<b>= 1</b>
<b>write(1, "i", 1i)</b>	<b>= 1</b>
<b>read(3, "s", 1)</b>	<b>= 1</b>
<b>write(1, "s", 1s)</b>	<b>= 1</b>
<b>read(3, "i", 1)</b>	<b>= 1</b>
<b>write(1, "i", 1i)</b>	<b>= 1</b>
<b>read(3, "o", 1)</b>	<b>= 1</b>
<b>write(1, "o", 1o)</b>	<b>= 1</b>
<b>read(3, "n", 1)</b>	<b>= 1</b>
<b>write(1, "n", 1n)</b>	<b>= 1</b>
<b>read(3, " ", 1)</b>	<b>= 1</b>
<b>write(1, " ", 1 )</b>	<b>= 1</b>
<b>read(3, "b", 1)</b>	<b>= 1</b>
<b>write(1, "b", 1b)</b>	<b>= 1</b>
<b>read(3, "y", 1)</b>	<b>= 1</b>
<b>write(1, "y", 1y)</b>	<b>= 1</b>
<b>read(3, " ", 1)</b>	<b>= 1</b>
<b>write(1, " ", 1 )</b>	<b>= 1</b>
<b>read(3, "z", 1)</b>	<b>= 1</b>
<b>write(1, "z", 1z)</b>	<b>= 1</b>
<b>read(3, "e", 1)</b>	<b>= 1</b>
<b>write(1, "e", 1e)</b>	<b>= 1</b>
<b>read(3, "r", 1)</b>	<b>= 1</b>
<b>write(1, "r", 1r)</b>	<b>= 1</b>
<b>read(3, "o", 1)</b>	<b>= 1</b>
<b>write(1, "o", 1o)</b>	<b>= 1</b>

```
read(3, ".", 1)          = 1
write(1, ".", 1)         = 1
read(3, "\n", 1)         = 1
write(1, "\n", 1
)                          = 1
read(3, "", 1)           = 0
close(3)                 = 0
wait4(-1, NULL, 0, NULL) = 125889
exit_group(0)            = ?
+++ exited with 0 +++
```

## Вывод

Данная лабораторная работа оказалась крайне необычной и мне до этого не приходилось использовать системные вызовы. Я научился использовать их, а также обмениваться данными между процессами используя каналы. Трудности возникли на моменте изучения и понимания материала. Не обошлось и без проблем при обработке ошибок системных вызовов, однако всё оказалось решаемым.