# Министерство науки и высшего образования РФ

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

# «Московский Авиационный Институт»

**Национальный Исследовательский Университет**

**Институт** №8 «Информационные технологии и прикладная математика»

**Кафедра** 806 «Вычислительная математика и программирование»

# Лабораторная работа №2

**по курсу «Операционные системы»**

|  |  |
| --- | --- |
| Студент: | Гуликов К.А. |
| Группа: | М8О-206Б-20 |
| Преподаватель: | Миронов Е. С. |
| Подпись: |  |
| Оценка: |  |
| Дата: |  |

Москва, 2022

# Содержание

1. Цель работы;
2. Постановка задачи;
3. Общие сведения о программе;
4. Общий метод и алгоритм решения;
5. Код программ;
6. Демонстрация работы программы;
7. Вывод.

# Цель работы

Приобретение практических навыков в:

* Управлении процессами в ОС;
* Обеспечении обмена данными между процессами посредством каналов.

# Постановка задачи

Составить и отладить программу на языке Си, осуществляющую работу с процессами и взаимодействие между ними в одной из двух операционных систем. В результате работы программа (основной процесс) должен создать для решение задачи один или несколько дочерних процессов. Взаимодействие между процессами осуществляется через системные сигналы/события и/или каналы (pipe).

Необходимо обрабатывать системные ошибки, которые могут возникнуть в результате работы.

**Вариант 8:** Родительский процесс создает дочерний процесс. Первой строчкой пользователь в консоль родительского процесса вводит имя файла, которое будет использовано для открытия файла с таким именем на чтение. Стандартный поток ввода дочернего процесса переопределяется открытым файлом. Дочерний процесс читает команды из стандартного потока ввода. Стандартный поток вывода дочернего процесса перенаправляется в pipe1. Родительский процесс читает из pipe1 и прочитанное выводит в свой стандартный поток вывода. Родительский и дочерний процесс должны быть представлены разными программами.

В файле записаны команды вида: «число число число<endline>». 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 int.

# Общие сведения о программе

Программа состоит из двух файлов: main1.c и main2.c. В данных файлах используются заголовочные файлы: stdio.h, stdlib.h, string.h, unistd.h.

Программы используют следующие системные вызовы:

1. read – читает данные из файла в буфер;
2. write – записывает данные из буфера в файл;
3. pipe – создает канала, через который процессы смогут обмениваться информацией;
4. fork – создает дочерний процесс, идентичный родительскому;
5. execv – заменяет образ памяти процесса;
6. close – закрывает открытый файл.

# Общий метод и алгоритм решения

* Читать имя файла для ввода как аргумент, создать дочерний процесс(с помощью fork) и pipe(для передачи данных из родительского процесса в дочерний), обработать возможные ошибки.
* Для родительского процесса: открыть файл, запустить программу с дочерним процессом. Позже вывести данные, которые передаст дочерний процесс после их обработки.
* Для дочернего процесса: считывать числа по символам, производить деление(при делении на 0 завершить работу дочернего и родительского процессов), передавать результат родительскому процессу через pipe, завершить работу.

# Код программ

**main1.c:**

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

#define PIPE\_FD\_SIZE 2 int main() {

FILE \*fp1 = NULL;

printf("Enter file name to read data: "); char \*name1=(char \*)malloc(256); scanf("%s",name1);

int fd1[PIPE\_FD\_SIZE]; if (pipe(fd1) == -1) {

printf("Can\'t create pipe!\n"); exit(0);

}

fp1 = freopen(name1, "r", stdin); if (fp1 == NULL) {

printf("Can\'t open file!\n"); exit(0);

}

int pr = fork(); if (pr == -1) {

printf("Can\'t fork child!\n"); exit(0);

} else if (pr == 0) { close(fd1[0]);

if (dup2(fd1[1], 1) == -1) { printf("Can\'t change stdout!\n"); exit(0);

}

char \* const \* argv = NULL; if (execv("ga2", argv) == -1) {

printf("Can\'t execute child process!\n"); exit(0);

}

} else {

close(fd1[1]); int res;

while (read(fd1[0], &res, sizeof(int))) { printf("%d\n", res);

}

close(fd1[0]);

}

return 0;

}

# main2.c:

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

#define PIPE\_FD\_SIZE 2

int main() {

char c;

int num = 0, res = 0, minus = 0, top = 1, pr = 0; while (scanf("%c", &c) > 0) {

if (c == ' ' || c == '\t') { if (num == 0) {

break;

} else {

if (top == 1)

top = 0;

else if (minus == 1)

res = (res / num) \* (-1); else res = res / num;

}

num = 0;

top = 0;

minus = 0;

pr = 1;

} else if (c == '-') { minus = 1;

pr = 0;

} else if (c == '\n') { if (num == 0) {

if (pr == 1) res = res / 1; else break;

} else {

if (top == 1)

top = 0;

else if (minus == 1)

res = (res / num) \* (-1); else res = res / num;

}

write(1, &res, sizeof(int)); num = 0;

top = 1;

minus = 0;

res = 0;

pr = 0;

} else if (c >= '0' && c <= '9') { num = num \*10 + c - '0';

if (top == 1)

res = num;

pr = 0;

}

}

}

# Демонстрация работы программы

konstantin@LAPTOP-44CRFC1U:~/labs/os/lab2$ cat test.txt 24 2 3

45 1 1 5

50 2 5

144 1 1 1 -1 1 1 2 1 3

108 1 1 1 1 5 1 1 1 3 1

144 1 1 1 1 1 12 1 1 1 1 1 2 1 1 1

44 4 1 1 1 1 0 1 1 1 11

konstantin@LAPTOP-44CRFC1U:~/labs/os/lab2$ ./main1 Enter file name to read data: test.txt

4

9

5

-24

7

6

konstantin@LAPTOP-44CRFC1U:~/labs/os/lab2$ cat test1.txt 78 2 1

77 11 1

210 2 1 1

108 5 1 1 1 1 1 1 2

210 2

450 1 5 1 0 1 1 1 2 1 1 3 1

102 1 1 1 2 1 1

konstantin@LAPTOP-44CRFC1U:~/labs/os/lab2$ ./main1 Enter file name to read data: test1.txt

39

7

105

10

105

# Вывод strace

konstantin@LAPTOP-44CRFC1U:~/labs/os/lab2$ strace ./ga1 < test.txt execve("./ga1", ["./ga1"], 0x7fffe198ad70 /\* 19 vars \*/) = 0

brk(NULL) = 0x7fffc4bbd000

access("/etc/ld.so.nohwcap", F\_OK) = -1 ENOENT (No such file or directory) 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=47603, ...}) = 0

mmap(NULL, 47603, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7f60228f5000

close(3) = 0

access("/etc/ld.so.nohwcap", F\_OK) = -1 ENOENT (No such file or directory) 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\260\34\2\0\0\0\0\0"..., 832) = 832 fstat(3, {st\_mode=S\_IFREG|0755, st\_size=2030544, ...}) = 0

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7f60228f0000

mmap(NULL, 4131552, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f6022200000

mprotect(0x7f60223e7000, 2097152, PROT\_NONE) = 0

mmap(0x7f60225e7000, 24576, PROT\_READ|PROT\_WRITE,

MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1e7000) = 0x7f60225e7000

mmap(0x7f60225ed000, 15072, PROT\_READ|PROT\_WRITE,

MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7f60225ed000

close(3) = 0

arch\_prctl(ARCH\_SET\_FS, 0x7f60228f14c0) = 0 mprotect(0x7f60225e7000, 16384, PROT\_READ) = 0

mprotect(0x7f6022c01000, 4096, PROT\_READ) = 0

mprotect(0x7f6022827000, 4096, PROT\_READ) = 0

munmap(0x7f60228f5000, 47603) = 0

fstat(1, {st\_mode=S\_IFCHR|0660, st\_rdev=makedev(4, 1), ...}) = 0 ioctl(1, TCGETS, {B38400 opost isig icanon echo ...}) = 0 brk(NULL) = 0x7fffc4bbd000

brk(0x7fffc4bde000) = 0x7fffc4bde000 pipe([3, 4]) = 0

clone(child\_stack=NULL, flags=CLONE\_CHILD\_CLEARTID|CLONE\_CHILD\_SETTID|SIGCHLD, child\_tidptr=0x7f60228f1790) = 109

close(4) = 0

read(3, "\4\0\0\0", 4) = 4

--- SIGCHLD {si\_signo=SIGCHLD, si\_code=CLD\_EXITED, si\_pid=109, si\_uid=1000, si\_status=0, si\_utime=0, si\_stime=0} ---

write(1, "Enter file name to read data: 4\n", 32Enter file name to read data: 4

) = 32

read(3, "\5\0\0\0", 4) = 4

write(1, "5\n", 25

) = 2

read(3, "\350\377\377\377", 4) = 4

write(1, "-24\n", 4-24

) = 4

read(3, "\6\0\0\0", 4) = 4

write(1, "6\n", 26

) = 2

read(3, "", 4) = 0

close(3) = 0

exit\_group(0) = ?

+++ exited with 0 +++

konstantin@LAPTOP-44CRFC1U:~/labs/os/lab2$ strace -T -i ./ga1 < test.txt [00007fa8bc0b4e97] execve("./ga1", ["./ga1"], 0x7ffff17887b0 /\* 19 vars \*/) = 0 <0.013370> [00007f14e7c1bef9] brk(NULL) = 0x7fffc09be000 <0.000045>

[00007f14e7c0f82e] access("/etc/ld.so.nohwcap", F\_OK) = -1 ENOENT (No such file or directory) <0.000282> [00007f14e7c1ce57] access("/etc/ld.so.preload", R\_OK) = -1 ENOENT (No such file or directory) <0.000241> [00007f14e7c1cd0d] openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC) = 3 <0.000269> [00007f14e7c1cc73] fstat(3, {st\_mode=S\_IFREG|0644, st\_size=47603, ...}) = 0 <0.000125> [00007f14e7c1cf73] mmap(NULL, 47603, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7f14e7ee4000 <0.000227>

[00007f14e7c1cf07] close(3) = 0 <0.000148>

[00007f14e7c18189] access("/etc/ld.so.nohwcap", F\_OK) = -1 ENOENT (No such file or directory) <0.000316> [00007f14e7c1cd0d] openat(AT\_FDCWD, "/lib/x86\_64-linux-gnu/libc.so.6", O\_RDONLY|O\_CLOEXEC) = 3

<0.000359>

[00007f14e7c1cdd4] read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\260\34\2\0\0\0\0\0"..., 832) = 832

<0.000122>

[00007f14e7c1cc73] fstat(3, {st\_mode=S\_IFREG|0755, st\_size=2030544, ...}) = 0 <0.000159>

[00007f14e7c1cf73] mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, - 1, 0) = 0x7f14e7ee0000 <0.000079>

[00007f14e7c1cf73] mmap(NULL, 4131552, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f14e7800000 <0.000281>

[00007f14e7c1d027] mprotect(0x7f14e79e7000, 2097152, PROT\_NONE) = 0 <0.000121>

[00007f14e7c1cf73] mmap(0x7f14e7be7000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1e7000) = 0x7f14e7be7000 <0.000177> [00007f14e7c1cf73] mmap(0x7f14e7bed000, 15072, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7f14e7bed000 <0.000150>

[00007f14e7c1cf07] close(3) = 0 <0.000226>

[00007f14e7c01024] arch\_prctl(ARCH\_SET\_FS, 0x7f14e7ee14c0) = 0 <0.000063>

[00007f14e7c1d027] mprotect(0x7f14e7be7000, 16384, PROT\_READ) = 0 <0.000074>

[00007f14e7c1d027] mprotect(0x7f14e8201000, 4096, PROT\_READ) = 0 <0.000150>

[00007f14e7c1d027] mprotect(0x7f14e7e27000, 4096, PROT\_READ) = 0 <0.000100>

[00007f14e7c1d007] munmap(0x7f14e7ee4000, 47603) = 0 <0.000298>

[00007f14e790f8d3] fstat(1, {st\_mode=S\_IFCHR|0660, st\_rdev=makedev(4, 1), ...}) = 0 <0.000139> [00007f14e7915da8] ioctl(1, TCGETS, {B38400 opost isig icanon echo ...}) = 0 <0.000092> [00007f14e79165b9] brk(NULL) = 0x7fffc09be000 <0.000043>

[00007f14e79165b9] brk(0x7fffc09df000) = 0x7fffc09df000 <0.000099> [00007f14e7910b17] pipe([3, 4]) = 0 <0.000139>

[00007f14e790fd9e] openat(AT\_FDCWD, "test.txt", O\_RDONLY) = 5 <0.000250> [00007f14e7910ae7] dup3(5, 0, 0) = 0 <0.000119>

[00007f14e79109e4] close(5) = 0 <0.000041>

[00007f14e78e4b7c] clone(child\_stack=NULL,

flags=CLONE\_CHILD\_CLEARTID|CLONE\_CHILD\_SETTID|SIGCHLD, child\_tidptr=0x7f14e7ee1790) = 89

<0.002425>

[00007f14e79109e4] close(4) = 0 <0.000104>

[00007f14e7910191] read(3, "\4\0\0\0", 4) = 4 <0.004019>

[00007f14e7910191] --- SIGCHLD {si\_signo=SIGCHLD, si\_code=CLD\_EXITED, si\_pid=89, si\_uid=1000, si\_status=0, si\_utime=0, si\_stime=0} ---

[00007f14e7910264] write(1, "Enter file name to read data: 4\n", 32Enter file name to read data: 4

) = 32 <0.000446>

[00007f14e7910191] read(3, "\5\0\0\0", 4) = 4 <0.000150> [00007f14e7910264] write(1, "5\n", 25

) = 2 <0.000436>

[00007f14e7910191] read(3, "\350\377\377\377", 4) = 4 <0.000142> [00007f14e7910264] write(1, "-24\n", 4-24

) = 4 <0.000499>

[00007f14e7910191] read(3, "\6\0\0\0", 4) = 4 <0.000067> [00007f14e7910264] write(1, "6\n", 26

) = 2 <0.000385>

[00007f14e7910191] read(3, "", 4) = 0 <0.000137>

[00007f14e79109e4] close(3) = 0 <0.000072>

[00007f14e78e4e66] exit\_group(0) = ? [????????????????] +++ exited with 0 +++

konstantin@LAPTOP-44CRFC1U:~/labs/os/lab2$ strace -r -f ./ga1 < test.txt 0.000000 execve("./ga1", ["./ga1"], 0x7fffd67bf6c0 /\* 19 vars \*/) = 0 0.004360 brk(NULL) = 0x7ffff7da8000

0.000796 access("/etc/ld.so.nohwcap", F\_OK) = -1 ENOENT (No such file or directory) 0.001117 access("/etc/ld.so.preload", R\_OK) = -1 ENOENT (No such file or directory) 0.001336 openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC) = 3

0.001188 fstat(3, {st\_mode=S\_IFREG|0644, st\_size=47603, ...}) = 0

0.001149 mmap(NULL, 47603, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7f3fc3629000

0.001070 close(3) = 0

0.002477 access("/etc/ld.so.nohwcap", F\_OK) = -1 ENOENT (No such file or directory)

0.001269 openat(AT\_FDCWD, "/lib/x86\_64-linux-gnu/libc.so.6", O\_RDONLY|O\_CLOEXEC) = 3

0.002975 read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\260\34\2\0\0\0\0\0"..., 832) = 832

0.001231 fstat(3, {st\_mode=S\_IFREG|0755, st\_size=2030544, ...}) = 0

0.001159 mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) =

0x7f3fc3620000

0.001138 mmap(NULL, 4131552, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) =

0x7f3fc2e00000

0.001073 mprotect(0x7f3fc2fe7000, 2097152, PROT\_NONE) = 0

0.001054 mmap(0x7f3fc31e7000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1e7000) = 0x7f3fc31e7000

0.001673 mmap(0x7f3fc31ed000, 15072, PROT\_READ|PROT\_WRITE,

MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7f3fc31ed000

0.001421 close(3) = 0

0.002784 arch\_prctl(ARCH\_SET\_FS, 0x7f3fc36214c0) = 0

0.001308 mprotect(0x7f3fc31e7000, 16384, PROT\_READ) = 0

0.002273 mprotect(0x7f3fc3a01000, 4096, PROT\_READ) = 0

0.001446 mprotect(0x7f3fc3427000, 4096, PROT\_READ) = 0

0.000947 munmap(0x7f3fc3629000, 47603) = 0

0.001174 fstat(1, {st\_mode=S\_IFCHR|0660, st\_rdev=makedev(4, 1), ...}) = 0 0.000795 ioctl(1, TCGETS, {B38400 opost isig icanon echo ...}) = 0 0.001023 brk(NULL) = 0x7ffff7da8000

0.000776 brk(0x7ffff7dc9000) = 0x7ffff7dc9000 0.001109 pipe([3, 4]) = 0

0.000913 openat(AT\_FDCWD, "test.txt", O\_RDONLY) = 5

0.001055 dup3(5, 0, 0) = 0

0.002430 close(5) = 0

0.001080 clone(child\_stack=NULL, flags=CLONE\_CHILD\_CLEARTID|CLONE\_CHILD\_SETTID|SIGCHLD,

child\_tidptr=0x7f3fc3621790) = 93 strace: Process 93 attached

|  |  |  |
| --- | --- | --- |
| [pid | 92] | 0.006459 close(4 <unfinished ...> |
| [pid | 93] | 0.001000 close(3 <unfinished ...> |
| [pid | 92] | 0.001000 <... close resumed> ) = 0 |
| [pid | 93] | 0.000946 <... close resumed> ) = 0 |
| [pid | 92] | 0.000856 read(3, <unfinished ...> |
| [pid | 93] | 0.000836 dup2(4, 1) = 1 |
| [pid | 93] | 0.001292 execve("ga2", NULL, 0x7ffffed54758 /\* 19 vars \*/) = 0 |
| [pid | 93] | 0.005516 brk(NULL) = 0x7fffc3335000 |
| [pid | 93] | 0.000975 access("/etc/ld.so.nohwcap", F\_OK) = -1 ENOENT (No such file or directory) |
| [pid | 93] | 0.003167 access("/etc/ld.so.preload", R\_OK) = -1 ENOENT (No such file or directory) |
| [pid | 93] | 0.001440 openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC) = 3 |
| [pid | 93] | 0.001307 fstat(3, {st\_mode=S\_IFREG|0644, st\_size=47603, ...}) = 0 |
| [pid | 93] | 0.001071 mmap(NULL, 47603, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7f56d4fce000 |
| [pid | 93] | 0.001036 close(3) = 0 |
| [pid | 93] | 0.000948 access("/etc/ld.so.nohwcap", F\_OK) = -1 ENOENT (No such file or directory) |
| [pid | 93] | 0.000999 openat(AT\_FDCWD, "/lib/x86\_64-linux-gnu/libc.so.6", O\_RDONLY|O\_CLOEXEC) = 3 |
| [pid | 93] | 0.002241 read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\260\34\2\0\0\0\0\0"..., 832) = 832 |
| [pid | 93] | 0.000931 fstat(3, {st\_mode=S\_IFREG|0755, st\_size=2030544, ...}) = 0 |
| [pid | 93] | 0.002929 mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, |

-1, 0) = 0x7f56d4fc0000

[pid 93] 0.000994 mmap(NULL, 4131552, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_DENYWRITE,

3, 0) = 0x7f56d4800000

[pid 93] 0.001005 mprotect(0x7f56d49e7000, 2097152, PROT\_NONE) = 0

[pid 93] 0.000998 mmap(0x7f56d4be7000, 24576, PROT\_READ|PROT\_WRITE,

MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1e7000) = 0x7f56d4be7000

[pid 93] 0.002111 mmap(0x7f56d4bed000, 15072, PROT\_READ|PROT\_WRITE,

MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7f56d4bed000

[pid 93] 0.001563 close(3) = 0

|  |  |  |
| --- | --- | --- |
| [pid | 93] | 0.000898 arch\_prctl(ARCH\_SET\_FS, 0x7f56d4fc1500) = 0 |
| [pid | 93] | 0.001232 mprotect(0x7f56d4be7000, 16384, PROT\_READ) = 0 |
| [pid | 93] | 0.000487 mprotect(0x7f56d5200000, 4096, PROT\_READ) = 0 |
| [pid | 93] | 0.000850 mprotect(0x7f56d4e27000, 4096, PROT\_READ) = 0 |
| [pid | 93] | 0.002816 munmap(0x7f56d4fce000, 47603) = 0 |
| [pid | 93] | 0.001266 fstat(0, {st\_mode=S\_IFREG|0644, st\_size=98, ...}) = 0 |
| [pid | 93] | 0.002659 brk(NULL) = 0x7fffc3335000 |
| [pid | 93] | 0.000848 brk(0x7fffc3356000) = 0x7fffc3356000 |
| [pid | 93] | 0.001052 read(0, "24 2 3\n50 2 5\n144 1 1 1 -1 1 1 2"..., 4096) = 98 |
| [pid | 93] | 0.001409 write(1, "\4\0\0\0", 4) = 4 |
| [pid | 92] | 0.001079 <... read resumed> "\4\0\0\0", 4) = 4 |
| [pid | 93] | 0.000701 write(1, "\5\0\0\0", 4 <unfinished ...> |
| [pid | 92] | 0.000892 write(1, "Enter file name to read data: 4\n", 32Enter file name to read data: 4 |
| <unfinished ...> | | |
| [pid | 93] | 0.001412 <... write resumed> ) = 4 |
| [pid | 92] | 0.000801 <... write resumed> ) = 32 |
| [pid | 93] | 0.000708 write(1, "\350\377\377\377", 4 <unfinished ...> |
| [pid | 92] | 0.000740 read(3, "\5\0\0\0", 4) = 4 |
| [pid | 93] | 0.002584 <... write resumed> ) = 4 |
| [pid | 92] | 0.000683 write(1, "5\n", 25 |
| <unfinished ...> | | |
| [pid | 93] | 0.003465 write(1, "\6\0\0\0", 4 <unfinished ...> |
| [pid | 92] | 0.000906 <... write resumed> ) = 2 |
| [pid | 93] | 0.000753 <... write resumed> ) = 4 |
| [pid | 92] | 0.001187 read(3, <unfinished ...> |
| [pid | 93] | 0.000916 lseek(0, -11, SEEK\_CUR <unfinished ...> |
| [pid | 92] | 0.000875 <... read resumed> "\350\377\377\377", 4) = 4 |
| [pid | 93] | 0.000764 <... lseek resumed> ) = 87 |
| [pid | 92] | 0.000695 write(1, "-24\n", 4-24 |

<unfinished ...>

|  |  |  |
| --- | --- | --- |
| [pid | 93] | 0.001321 exit\_group(0 <unfinished ...> |
| [pid | 92] | 0.000815 <... write resumed> ) = 4 |
| [pid | 92] | 0.002341 read(3, <unfinished ...> |
| [pid | 93] | 0.000705 <... exit\_group resumed>) = ? |
| [pid | 92] | 0.002520 <... read resumed> "\6\0\0\0", 4) = 4 |
| [pid | 92] | 0.000653 --- SIGCHLD {si\_signo=SIGCHLD, si\_code=CLD\_EXITED, si\_pid=93, si\_uid=1000, |

si\_status=0, si\_utime=0, si\_stime=0} ---

[pid 93] 0.001203 +++ exited with 0 +++

0.000672 write(1, "6\n", 26

) = 2

0.001722 read(3, "", 4) = 0

0.000892 close(3) = 0

0.001334 exit\_group(0) = ? 0.001476 +++ exited with 0 +++

# Замечания

Во время тестирования программы обнаружилась ошибка, которая состояла в наличии пробела в тестовом файле перед переносом строки. Программа исправлена, с учетом проверки предпоследнего символа строки на пробел и работает без ошибок.

# Вывод

В результате выполнения данной лабораторной работы я научился работать с процессами, реализовывать обмен информацией между дочерним и родительским процессом, запускать одну программу из другой, а также работать с strace.

В программе межпроцессорное взаимодействие осуществляется с помощью канала. На мой взгляд, такой способ общения процессов очень удобен, так как при данном подходе реже приходится сталкиваться с гонками, потому что при использовании блокирующих системных вызовов read и write процессы блокируются, если им нечего считывать или буфер для записи полный. Так же одним из плюсов такого подхода к межпроцессорному взаимодействию является то, что неименованным каналом могут пользоваться только родственные процессы, так как канал находится в пределах ядра.