Московский Авиационный Институт
(Национальный Исследовательский Университет)
Факультет информационных технологий и прикладной математики
Кафедра вычислительной математики и программирования

Лабораторная работа №1 по курсу

«Операционные системы»

Студент: Пирогов М.Д.
Группа: М8О-207Б-21
Преподаватель: Миронов Евгений Сергеевич
Оценка: _____
Дата: _____
Подпись: _____

Москва, 2022

Содержание

Репозиторий

https://github.com/pirogovmark/OS-Labs

Постановка задачи

Цель работы

Приобретение практических навыков диагностики работы программного обеспечения.

Задание

При выполнении последующих лабораторных работ необходимо продемонстрировать ключевые системные вызовы, которые в них используются и то, что их использование соответствует варианту ЛР.

Проведу диагностику для второй ЛР.

Общие сведения о программе

Для диагностики работы программного обеспечения используется утилита strace.

Системные вызовы:

1. arch_prctl - установить состояние треда, специфичное для архитектуры
2. madvise - выдает предложения об использовании памяти
3. exit - обычное завершение работы программы
4. access - проверить права доступа пользователя к файлу
5. openat, open – открывает файл
6. mmap, munmap - отражает файлы или устройства в памяти или снимает их отражение
7. stat, fstat, lstat - считывает статус файла
8. brk, sbrk - изменение размера сегмента данных
9. execve - выполняет программу, заданную параметром *filename*
10. pipe - создает канал
11. clone - создать процесс-потомок
12. lseek - установить смещение для позиционирования операций чтения/записи
13. futex - системный вызов быстрых связей пространства пользователя

void *mmap(void *addr, size_t length, int prot, int flags, int fd, off_t offset) – возвращает указатель на начало выделенного блока памяти. Addr — позволяет выбрать конкретный адрес, length — длина участвка, int prot — разрешения (write, read), fd — файловый дескриптор, offset — сдвиг относительно адреса.

int access(const char * pathname, int mode) – проверяет, имеет ли процесс права на чтение или запись, или же просто проверяет, существует ли файл (или другой объект файловой системы), с

именем pathname. Если pathname является символьной ссылкой, то проверяются права доступа к файлу, на который она ссылается.

mode -- это маска, состоящая из одного или более флагов **R_OK**, **W_OK**, **X_OK** и **F_OK**. **R_OK**, **W_OK** и **X_OK** запрашивают соответственно проверку существования файла и возможности его чтения, записи или выполнения. **F_OK** просто проверяет существование файла.

int pipe(int filedes[2]) – создает пару файловых описателей, указывающих на запись inode именованного канала, и помещает их в массив, на который указывает fildes. fildes[0] предназначен для чтения, а fildes[1] предназначен для записи.

int open(const char *pathname, int flags) - вызов **open()** используется, чтобы преобразовать путь к файлу в описатель файла (небольшое неотрицательное целое число, которое используется с вызовами **read**, **write** и т.п. при последующем вводе-выводе). Если системный вызов завершается успешно, возвращенный файловый описатель является наименьшим описателем, который еще не открыт процессом. В результате этого вызова появляется новый открытый файл, не разделяемый никакими процессами. Новый описатель файла будет оставаться открытым при выполнении функции exec. Указатель устанавливается в начале файла. Параметр *flags* - это флаги **O_RDONLY**, **O_WRONLY** или **O_RDWR**, открывающие файлы "только для чтения", "только для записи" и для чтения и записи соответственно.

Демонстрация работы программы

```
markm1@ubuntum1:~/Downloads/Lab_2/build$ strace -f ./main

execve("./main", ["./main"], 0xfffffdad7b38 /* 46 vars */) = 0

brk(NULL)                               = 0xaaab1b787000

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0xffffa2b42000

faccessat(AT_FDCWD, "/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=63061, ...}, AT_EMPTY_PATH) = 0

mmap(NULL, 63061, PROT_READ, MAP_PRIVATE, 3, 0) = 0xffffa2b32000

close(3)                                = 0

openat(AT_FDCWD, "/lib/aarch64-linux-gnu/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=2239880, ...}, AT_EMPTY_PATH) = 0
```

mmap(NULL, 2316320, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffffa28d3000

mmap(0xffffa28e0000, 2250784, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffffa28e0000

munmap(0xffffa28d3000, 53248) = 0

munmap(0xffffa2b06000, 10272) = 0

mprotect(0xffffa2ae8000, 53248, PROT_NONE) = 0

mmap(0xffffa2af5000, 57344, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x215000) = 0xffffa2af5000

mmap(0xffffa2b03000, 10272, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0xffffa2b03000

close(3) = 0

openat(AT_FDCWD, "/lib/aarch64-linux-gnu/libgcc_s.so.1", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=133448, ...}, AT_EMPTY_PATH) = 0

mmap(NULL, 262856, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffffa289f000

mmap(0xffffa28a0000, 197320, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffffa28a0000

munmap(0xffffa289f000, 4096) = 0

munmap(0xffffa28d1000, 58056) = 0

mprotect(0xffffa28b4000, 110592, PROT_NONE) = 0

mmap(0xffffa28cf000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1f000) = 0xffffa28cf000

close(3) = 0

openat(AT_FDCWD, "/lib/aarch64-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\267\0\1\0\0\0py\2\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=1657920, ...}, AT_EMPTY_PATH) = 0

mmap(NULL, 1826976, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffffa26e1000

mmap(0xffffa26f0000, 1761440, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffffa26f0000

munmap(0xffffa26e1000, 61440)   = 0

munmap(0xffffa289f000, 160)   = 0

mprotect(0xffffa287c000, 65536, PROT_NONE) = 0

mmap(0xffffa288c000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x18c000) = 0xffffa288c000

mmap(0xffffa2892000, 49312, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0xffffa2892000

close(3)   = 0

openat(AT_FDCWD, "/lib/aarch64-linux-gnu/libm.so.6", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=592024, ...}, AT_EMPTY_PATH) = 0

mmap(NULL, 721008, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffffa263f000

mmap(0xffffa2640000, 655472, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffffa2640000

munmap(0xffffa263f000, 4096)   = 0

munmap(0xffffa26e1000, 57456)   = 0

mprotect(0xffffa26c3000, 114688, PROT_NONE) = 0

mmap(0xffffa26df000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x8f000) = 0xffffa26df000

close(3)   = 0

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0xffffa2b07000

set_tid_address(0xffffa2b07b10)   = 2257

set_robust_list(0xffffa2b07b20, 24)   = 0

rseq(0xffffa2b08160, 0x20, 0, 0xd428bc00) = 0

mprotect(0xffffa288c000, 16384, PROT_READ) = 0

mprotect(0xffffa26df000, 4096, PROT_READ) = 0

mprotect(0xffffa28cf000, 4096, PROT_READ) = 0

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffffa28de000

mprotect(0xffffa2af5000, 45056, PROT_READ) = 0

mprotect(0xaaaae5fef000, 4096, PROT_READ) = 0

mprotect(0xffffa2b47000, 8192, PROT_READ) = 0

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

munmap(0xffffa2b32000, 63061) = 0

getrandom("\x5a\xfd\x38\x21\x05\x0d\x3e\x36", 8, GRND_NONBLOCK) = 8

brk(NULL) = 0xaaab1b787000

brk(0xaaab1b7a8000) = 0xaaab1b7a8000

futex(0xffffa2b037a4, FUTEX_WAKE_PRIVATE, 2147483647) = 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|SIGCHLDstrace: Process 2258 attached

, child_tidptr=0xffffa2b07b10) = 2258

[pid 2257] clone(child_stack=NULL, flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLD <unfinished ...>

[pid 2258] set_robust_list(0xffffa2b07b20, 24) = 0

[pid 2257] <... clone resumed>, child_tidptr=0xffffa2b07b10) = 2259

strace: Process 2259 attached

[pid 2258] close(7 <unfinished ...>

[pid 2257] close(5 <unfinished ...>

[pid 2258] <... close resumed>) = 0

[pid 2259] set_robust_list(0xffffa2b07b20, 24 <unfinished ...>

[pid 2257] <... close resumed>) = 0

[pid 2259] <... set_robust_list resumed>) = 0

[pid 2258] close(8 <unfinished ...>

[pid 2257] close(6 <unfinished ...>

[pid 2258] <... close resumed>)   = 0

[pid 2257] <... close resumed>)   = 0

[pid 2259] close(3 <unfinished ...>

[pid 2258] close(4)   = 0

[pid 2258] close(5)   = 0

[pid 2257] close(3 <unfinished ...>

[pid 2259] <... close resumed>)   = 0

[pid 2258] read(3, <unfinished ...>

[pid 2257] <... close resumed>)   = 0

[pid 2259] close(4 <unfinished ...>

[pid 2257] close(8 <unfinished ...>

[pid 2259] <... close resumed>)   = 0

[pid 2257] <... close resumed>)   = 0

[pid 2259] close(6)   = 0

[pid 2257] newfstatat(1, "", <unfinished ...>

[pid 2259] close(7)   = 0

[pid 2257] <... newfstatat resumed>{st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}, AT_EMPTY_PATH) = 0

[pid 2259] read(5, <unfinished ...>

[pid 2257] write(1, "Enter the number of lines: ", 27Enter the number of lines:) = 27

[pid 2257] newfstatat(0, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}, AT_EMPTY_PATH) = 0

[pid 2257] read(0, 4

"4\n", 1024)   = 2

[pid 2257] read(0, ADbbb   asdasd   asdasd

"ADbbb   asdasd   asdasd\n", 1024) = 24

[pid 2257] read(0, ASASD   asdasd asd

"ASASD   asdasd asd\n", 1024) = 20

[pid 2257] read(0, ASAS   asdasd   asd

"ASAS   asdasd   asd\n", 1024) = 20

[pid 2257] read(0, ASDAS   asdasdasd asdasd

"ASDAS   asdasdasd asdasd\n", 1024) = 25

[pid 2257] write(1, "\nADbbb   asdasd   asdasd\nASASD "..., 90

ADbbb   asdasd   asdasd

ASASD   asdasd asd

ASAS   asdasd   asd

ASDAS   asdasdasd asdasd

) = 90

[pid 2257] write(1, "\n", 1

)   = 1

[pid 2257] write(1, "Parent in: \nADbbb   asdasd   asd"..., 101Parent in:

ADbbb   asdasd   asdasd

ASASD   asdasd asd

ASAS   asdasd   asd

ASDAS   asdasdasd asdasd

) = 101

[pid 2257] write(1, "\n\n", 2

)   = 2

[pid 2257] write(4, "\nADbbb   asdasd   asdasd\nASASD "..., 90) = 90

[pid 2257] close(4 <unfinished ...>

[pid 2258] <... read resumed>"\n", 1) = 1

[pid 2257] <... close resumed>)   = 0

[pid 2258] read(3, <unfinished ...>

[pid 2257] read(7, <unfinished ...>

[pid 2258] <... read resumed>"A", 1)   = 1

[pid 2258] read(3, "D", 1)   = 1

[pid 2258] read(3, "b", 1)   = 1

[pid 2258] read(3, "b", 1)   = 1

[pid 2258] read(3, "b", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, "\n", 1)   = 1

[pid 2258] read(3, "A", 1)   = 1

[pid 2258] read(3, "S", 1)   = 1

[pid 2258] read(3, "A", 1)   = 1

[pid 2258] read(3, "S", 1)   = 1

[pid 2258] read(3, "D", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, "\n", 1)   = 1

[pid 2258] read(3, "A", 1)   = 1

[pid 2258] read(3, "S", 1)   = 1

[pid 2258] read(3, "A", 1)   = 1

[pid 2258] read(3, "S", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, "\n", 1)   = 1

[pid 2258] read(3, "A", 1)   = 1

[pid 2258] read(3, "S", 1)   = 1

[pid 2258] read(3, "D", 1)   = 1

[pid 2258] read(3, "A", 1)   = 1

[pid 2258] read(3, "S", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, " ", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, "a", 1)   = 1

[pid 2258] read(3, "s", 1)   = 1

[pid 2258] read(3, "d", 1)   = 1

[pid 2258] read(3, "\n", 1)   = 1

[pid 2258] read(3, "", 1)   = 0

[pid 2258] newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}, AT_EMPTY_PATH) = 0

[pid 2258] write(1, "Child 1 in: \nADbbb   asdasd   as"..., 102Child 1 in:

ADbbb   asdasd   asdasd

ASASD   asdasd asd

ASAS   asdasd   asd

ASDAS   asdasdasd asdasd

) = 102

[pid 2258] write(1, "\n", 1

)   = 1

[pid 2258] close(3)   = 0

[pid 2258] write(1, "Child 1 out: \nADBBB   ASDASD   A"..., 103Child 1 out:

ADBBB   ASDASD   ASDASD

ASASD   ASDASD ASD

ASAS   ASDASD   ASD

ASDAS   ASDASDASD ASDASD

) = 103

[pid 2258] write(1, "\n", 1

)   = 1

[pid 2258] write(6, "\nADBBB   ASDASD   ASDASD\nASASD "..., 90 <unfinished ...>

[pid 2259] <... read resumed>"\n", 1) = 1

[pid 2258] <... write resumed>)   = 90

[pid 2259] read(5, <unfinished ...>

[pid 2258] close(6 <unfinished ...>

[pid 2259] <... read resumed>"A", 1)   = 1

[pid 2258] <... close resumed>)   = 0

[pid 2259] read(5, <unfinished ...>

[pid 2258] write(1, "\n", 1

 <unfinished ...>

[pid 2259] <... read resumed>"D", 1)   = 1

[pid 2258] <... write resumed>)   = 1

[pid 2259] read(5, <unfinished ...>

[pid 2258] exit_group(0 <unfinished ...>

[pid 2259] <... read resumed>"B", 1)   = 1

[pid 2258] <... exit_group resumed>)   = ?

[pid 2259] read(5, <unfinished ...>

[pid 2258] +++ exited with 0 +++

[pid 2257] <... read resumed>0xfffff05f6f88, 1) = ? ERESTARTSYS (To be restarted if SA_RESTART is set)

[pid 2259] <... read resumed>"B", 1)   = 1

[pid 2257] --- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=2258, si_uid=1000, si_status=0, si_utime=0, si_stime=0} ---

```
[pid  2259] read(5,  <unfinished ...>
[pid  2257] read(7,  <unfinished ...>
[pid  2259] <... read resumed>"B", 1)   = 1
[pid  2259] read(5, " ", 1)             = 1
[pid  2259] read(5, " ", 1)             = 1
[pid  2259] read(5, " ", 1)             = 1
[pid  2259] read(5, "A", 1)             = 1
[pid  2259] read(5, "S", 1)             = 1
[pid  2259] read(5, "D", 1)             = 1
[pid  2259] read(5, "A", 1)             = 1
[pid  2259] read(5, "S", 1)             = 1
[pid  2259] read(5, "D", 1)             = 1
[pid  2259] read(5, " ", 1)             = 1
[pid  2259] read(5, " ", 1)             = 1
[pid  2259] read(5, " ", 1)             = 1
[pid  2259] read(5, "A", 1)             = 1
[pid  2259] read(5, "S", 1)             = 1
[pid  2259] read(5, "D", 1)             = 1
[pid  2259] read(5, "A", 1)             = 1
[pid  2259] read(5, "S", 1)             = 1
[pid  2259] read(5, "D", 1)             = 1
[pid  2259] read(5, "\n", 1)            = 1
[pid  2259] read(5, "A", 1)             = 1
[pid  2259] read(5, "S", 1)             = 1
[pid  2259] read(5, "A", 1)             = 1
[pid  2259] read(5, "S", 1)             = 1
[pid  2259] read(5, "D", 1)             = 1
```

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, "\n", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, "\n", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, " ", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, "A", 1)   = 1

[pid 2259] read(5, "S", 1)   = 1

[pid 2259] read(5, "D", 1)   = 1

[pid 2259] read(5, "\n", 1)   = 1

[pid 2259] read(5, "", 1)   = 0

[pid 2259] newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}, AT_EMPTY_PATH) = 0

[pid 2259] write(1, "Child 2 in: \nADBBB   ASDASD   AS"..., 102Child 2 in:

ADBBB   ASDASD   ASDASD

ASASD   ASDASD ASD

ASAS   ASDASD   ASD

ASDAS   ASDASDASD ASDASD

) = 102

[pid 2259] write(1, "\n", 1

)   = 1

[pid 2259] close(5)   = 0

[pid 2259] write(1, "Child 2 out: ADBBB ASDASD ASDASD"..., 89Child 2 out: ADBBB ASDASD ASDASD

ASASD ASDASD ASD

ASAS ASDASD ASD

ASDAS ASDASDASD ASDASD

) = 89

[pid 2259] write(1, "\n", 1

)   = 1

[pid 2259] write(8, "ADBBB ASDASD ASDASD\nASASD ASDASD"..., 76) = 76

[pid 2257] <... read resumed>"A", 1)   = 1

[pid 2257] read(7, <unfinished ...>

[pid 2259] close(8 <unfinished ...>

[pid 2257] <... read resumed>"D", 1)   = 1

[pid 2259] <... close resumed>)   = 0

[pid 2257] read(7, <unfinished ...>

[pid 2259] write(1, "\n", 1

 <unfinished ...>

[pid 2257] <... read resumed>"B", 1)   = 1

[pid 2259] <... write resumed>)   = 1

[pid 2257] read(7, "B", 1)   = 1

[pid 2257] read(7, <unfinished ...>

[pid 2259] exit_group(0 <unfinished ...>

[pid 2257] <... read resumed>"B", 1)   = 1

[pid 2257] read(7, <unfinished ...>

[pid 2259] <... exit_group resumed>)   = ?

[pid 2257] <... read resumed>" ", 1)   = 1

[pid 2259] +++ exited with 0 +++

--- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=2259, si_uid=1000, si_status=0, si_utime=0, si_stime=0} ---

read(7, "A", 1)   = 1

read(7, "S", 1)   = 1

read(7, "D", 1)   = 1

read(7, "A", 1)   = 1

read(7, "S", 1)   = 1

read(7, "D", 1)   = 1

read(7, " ", 1)   = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, "\n", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, " ", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, " ", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, "\n", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, " ", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, " ", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, "\n", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, " ", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

read(7, "D", 1) = 1

read(7, "A", 1) = 1

read(7, "S", 1) = 1

```
read(7, "D", 1)                 = 1
read(7, " ", 1)                 = 1
read(7, "A", 1)                 = 1
read(7, "S", 1)                 = 1
read(7, "D", 1)                 = 1
read(7, "A", 1)                 = 1
read(7, "S", 1)                 = 1
read(7, "D", 1)                 = 1
read(7, "\n", 1)                = 1
read(7, "", 1)                  = 0
write(1, "Parent out: ADBBB ASDASD ASDASD\n"..., 88Parent out: ADBBB ASDASD ASDASD
ASASD ASDASD ASD
ASAS ASDASD ASD
ASDAS ASDASDASD ASDASD
) = 88
write(1, "\n", 1
)                       = 1
close(7)                        = 0
exit_group(0)                   = ?
+++ exited with 0 +++
```

Выводы

Данная лабораторная работа оказалась полезной. Я приобрел практические навыки в диагностике работы программного обеспечения.