Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

О Т Ч Е Т

по лабораторной работе № 0 4

Дисциплина: ***Операционные системы***

Студент	***ИУ7И-66Б***		**Нгуен Ф. С.**
	(Группа)	(Подпись, дата)	(И.О. Фамилия)
Преподаватель			**Рязанова Н. Ю.**
		(Подпись, дата)	(И.О. Фамилия)

Москва, 2021

Код программы

```
void daemonize(const char *cmd)
{
    int i, fd0, fd1, fd2;
    pid_t pid;
    struct rlimit rl;
    struct sigaction sa;

    umask(0);


    if (getrlimit(RLIMIT_NOFILE, &rl) < 0)
        err_quit("%s: невозможно получить максимальный номер дескриптора ", cmd);


    if ((pid = fork()) < 0)
        err_quit("%s: ошибка вызова функции fork", cmd);
    else if (pid != 0)
        exit(0);
    setsid();


    sa.sa_handler = SIG_IGN;
    sigemptyset(&sa.sa_mask);
    sa.sa_flags = 0;
    if (sigaction(SIGHUP, &sa, NULL) < 0)
        err_quit("%s: невозможно игнорировать сигнал SIGHUP", cmd);
    if ((pid = fork()) < 0)
        err_quit("%s: ошибка вызова функции fork", cmd);
    else if (pid != 0)
        exit(0);


    if (chdir("/") < 0)
        err_quit("%s: невозможно сделать текущим рабочим каталогом /", cmd);


    if (rl.rlim_max == RLIM_INFINITY)
        rl.rlim_max = 1024;
    for (i = 0; i < rl.rlim_max; i++)
        close(i);


    fd0 = open("/dev/null", O_RDWR);
    fd1 = dup(0);
    fd2 = dup(0);


    if (fd0 != 0 || fd1 != 1 || fd2 != 2)
    {
        syslog(LOG_ERR, "ошибочные файловые дескрипторы %d %d %d",
               fd0, fd1, fd2);
        exit(1);
    }
}

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

#define LOG_FILE "/home/nguyensang/Desktop/OS2021/lab4/part1/log.txt"
#define BUF_SIZE 0x100
```

```
struct print_info
{
    char name[NAME_MAX];
    int ignore_newline;
    void (*print_file)(FILE *);
    void (*print_name)(char *);
};

void *get_proc_filename(char *name, int pid, const char *shortname);
void print_proc_file(int pid, struct print_info *info);
void print_proc_stat(FILE *);
void print_proc_statm(FILE *);
void print_proc_mem(FILE *);
void print_proc_fd(char *);
void print_symlink(char *);
void print_fields(FILE *, char **fields);

FILE *f;

int main(int argc, char *argv[])
{
    daemonize("DAEMON");

    f = fopen(LOG_FILE, "w");

    char buf[BUF_SIZE];
    int pid;

    struct print_info files[][8] = {
        {"cmdline"},
        {"cwd", 0, NULL, print_symlink},
        {"environ"},
        {"exe", 0, NULL, print_symlink},
        {"fd", 0, NULL, print_proc_fd},
        {"maps"},
        {"mem"},
        {"root", 0, NULL, print_symlink},
        {"stat", 0, print_proc_stat},
        {"statm", 0, print_proc_statm},
    };

    int n_files = sizeof(files) / sizeof(files[0]);

    if (argc != 2 || (pid = atoi(argv[1])) <= 0)
        pid = getpid();

    fprintf(f, "PID = %d\n", pid);

    for (int i = 0; i < n_files; i++)
    {
        print_proc_file(pid, files[i]);
    }

    fclose(f);

    return 0;
}

void *get_proc_filename(char *name, int pid, const char *shortname)
{
    sprintf(name, "/proc/%d/%s", pid, shortname);
}

void print_proc_file(int pid, struct print_info *info)
{
    fprintf(f, "\n[--%s--]\n", info->name);
```

```
    char name[PATH_MAX];

    get_proc_filename(name, pid, info->name);



    if (info->print_name)
        return info->print_name(name);

    FILE *file = fopen(name, "r");
    if (!file)
        return;

    if (info->print_file)
    {
        info->print_file(file);
        fclose(file);
        return;
    }

    char buf[BUF_SIZE];
    int len, i;

    while ((len = fread(buf, 1, BUF_SIZE, file)) > 0)
    {
        if (!info->ignore_newline)
            for (i = 0; i < len; i++)
                if (buf[i] == 0)
                    buf[i] = 10;

        buf[len] = 0;
        fprintf(f, "%s", buf);
    }

    fclose(file);
}

void print_proc_stat(FILE *file)
{
    static char *fields[] = {
        "pid", "comm", "state", "ppid", "pgrp", "session", "tty_nr", "tpgid",
        "flags", "minflt", "cminflt", "majflt", "cmajflt", "utime", "stime",
        "cutime", "cstime", "priority", "nice", "num threads", "itrealvalue",
        "starttime", "vsize", "rss", "rsslim", "startcode", "endcode",
        "startstack", "kstkesp", "kstkeip", "signal", "blocked", "sigignore",
        "sigcatch", "wchan", "nswap", "cnswap", "exit_signal", "processor",
        "rt_priority", "policy", "delayacct_blkio_ticks", "guest_time",
        "cguest_time", "start_data", "end_data", "start_brk", "arg_start",
        "arg_end", "env_start", "env_end", "exit_code", NULL};

    print_fields(file, fields);
}

void print_proc_statm(FILE *file)
{
    static char *fields[] = {
        "size", "resident", "shared", "text", "lib", "data", "dt", NULL};

    print_fields(file, fields);
}

void print_proc_fd(char *name)
{
    DIR *dir = opendir(name);
    if (!dir)
    {
```

```
        fprintf(stderr, "opendir('%s'): %s\n", name, strerror(errno));
        exit(EXIT_FAILURE);
    }

    int flag = 1;
    char path[PATH_MAX];

    struct dirent *dirp = NULL;
    while (flag && (dirp = readdir(dir)))
    {
        if (strcmp(dirp->d_name, ".") == 0)
        {
            flag = 1;
        }
        else if (strcmp(dirp->d_name, "..") == 0)
        {
            flag = 1;
        }
        else
        {
            snprintf(path, sizeof(path), "%s/%s", name, dirp->d_name);
            print_symlink(path);
        }
    }

    closedir(dir);
}

void print_symlink(char *path)
{
    char str[PATH_MAX];
    const int n = readlink(path, str, sizeof(str));
    str[n] = '\0';

    fprintf(f, "%s\t->\t%s\n", path, str);
}

void print_fields(FILE *file, char **fields)
{
    char buf[BUF_SIZE];
    const size_t len = fread(buf, 1, BUF_SIZE, file);
    buf[len - 1] = '\0';

    char *value, **pfield;

    for (
        value = strtok(buf, " "), pfield = fields;
        value && *pfield; ++pfield)
    {
        fprintf(f, "%-22s %s\n", *pfield, value);
        value = strtok(NULL, " ");
    }
}

void print_proc_mem(FILE *file)
{
    char buf[BUF_SIZE];
    int len, i;

    while ((len = fread(buf, 1, BUF_SIZE, file)) > 0)
    {
        for (i = 0; i < len; i++)
            if (buf[i] == 0)
                buf[i] = 10; // '\n'

        buf[len] = 0;
        fprintf(f, "%s", buf);
```

```
    }
}
```

Результат

```
PID = 4029

[--cmdline--]
./a.out

[--cwd--]
/proc/4029/cwd  ->  /

[--environ--]
SHELL=/bin/bash
SESSION_MANAGER=local/K-virtual-machine:@/tmp/.ICE-unix/1698,unix/K-virtual-
machine:/tmp/.ICE-unix/1698
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
LC_ADDRESS=ru_RU.UTF-8
GNOME_SHELL_SESSION_MODE=ubuntu
LC_NAME=ru_RU.UTF-8
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
XMODIFIERS=@im=ibus
DESKTOP_SESSION=ubuntu
LC_MONETARY=ru_RU.UTF-8
SSH_AGENT_PID=1664
GTK_MODULES=gail:atk-bridge
DBUS_STARTER_BUS_TYPE=session
PWD=/home/nguyensang/Desktop/OS2021/lab4/part1
LOGNAME=nguyensang
XDG_SESSION_DESKTOP=ubuntu
XDG_SESSION_TYPE=x11
GPG_AGENT_INFO=/run/user/1000/gnupg/S.gpg-agent:0:1
XAUTHORITY=/run/user/1000/gdm/Xauthority
WINDOWPATH=2
HOME=/home/nguyensang
USERNAME=nguyensang
IM_CONFIG_PHASE=1
LC_PAPER=ru_RU.UTF-8
LANG=en_US.UTF-8
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;01:cd=
40;33;01:or=40;31;01:mi=00:su=37;41:sg=30;43:ca=30;41:tw=30;42:ow=34;42:st=37;44:
ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.taz=01;31:*.lha=01;31:
*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.tzo=01;31:*.t7z=01
;31:*.zip=01;31:*.z=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31:*.lz=01;31:*.lzo=01;3
1:*.xz=01;31:*.zst=01;31:*.tzst=01;31:*.bz2=01;31:*.bz=01;31:*.tbz=01;31:*.tbz2=0
1;31:*.tz=01;31:*.deb=01;31:*.rpm=01;31:*.jar=01;31:*.war=01;31:*.ear=01;31:*.sar
=01;31:*.rar=01;31:*.alz=01;31:*.ace=01;31:*.zoo=01;31:*.cpio=01;31:*.7z=01;31:*.
rz=01;31:*.cab=01;31:*.wim=01;31:*.swm=01;31:*.dwm=01;31:*.esd=01;31:*.jpg=01;35:
*.jpeg=01;35:*.mjpg=01;35:*.mjpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm
=01;35:*.ppm=01;35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*
.png=01;35:*.svg=01;35:*.svgz=01;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;
35:*.mpeg=01;35:*.m2v=01;35:*.mkv=01;35:*.webm=01;35:*.ogm=01;35:*.mp4=01;35:*.m4
v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.nuv=01;35:*.wmv=01;35:*.asf=01;35:*
.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.fli=01;35:*.flv=01;35:*.gl=01;35
:*.dl=01;35:*.xcf=01;35:*.xwd=01;35:*.yuv=01;35:*.cgm=01;35:*.emf=01;35:*.ogv=01;
35:*.ogx=01;35:*.aac=00;36:*.au=00;36:*.flac=00;36:*.m4a=00;36:*.mid=00;36:*.midi
=00;36:*.mka=00;36:*.mp3=00;36:*.mpc=00;36:*.ogg=00;36:*.ra=00;36:*.wav=00;36:*.o
ga=00;36:*.opus=00;36:*.spx=00;36:*.xspf=00;36:
XDG_CURRENT_DESKTOP=ubuntu:GNOME
```

```
VTE_VERSION=6003
GNOME_TERMINAL_SCREEN=/org/gnome/Terminal/screen/ce80e0fd_fd1a_4276_a22f_073ca97a
2080
INVOCATION_ID=9058a45c78f247548a40620067e93b07
MANAGERPID=1483
LESSCLOSE=/usr/bin/lesspipe %s %s
XDG_SESSION_CLASS=user
TERM=xterm-256color
LC_IDENTIFICATION=ru_RU.UTF-8
LESSOPEN=| /usr/bin/lesspipe %s
USER=nguyensang
GNOME_TERMINAL_SERVICE=:1.82
DISPLAY=:0
SHLVL=1
LC_TELEPHONE=ru_RU.UTF-8
QT_IM_MODULE=ibus
LC_MEASUREMENT=ru_RU.UTF-8
DBUS_STARTER_ADDRESS=unix:path=/run/user/1000/bus,guid=c34295a75cc445ae7ee4d54860
cf8253
XDG_RUNTIME_DIR=/run/user/1000
LC_TIME=ru_RU.UTF-8
JOURNAL_STREAM=8:53207
XDG_DATA_DIRS=/usr/share/ubuntu:/usr/local/share/:/usr/share/:/var/lib/snapd/desk
top
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr
/local/games:/snap/bin
GDMSESSION=ubuntu
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus,guid=c34295a75cc445ae7ee4d5
4860cf8253
LC_NUMERIC=ru_RU.UTF-8
_=./a.out

[--exe--]
/proc/4029/exe  ->  /home/nguyensang/Desktop/OS2021/lab4/part1/a.out

[--fd--]
/proc/4029/fd/0 ->  /dev/null
/proc/4029/fd/1 ->  /dev/null
/proc/4029/fd/2 ->  /dev/null
/proc/4029/fd/3 ->  /home/nguyensang/Desktop/OS2021/lab4/part1/log.txt
/proc/4029/fd/4 ->  /proc/4029/fd

[--maps--]
55cd54027000-55cd54029000 r--p 00000000 08:05 2228632
/home/nguyensang/Desktop/OS2021/lab4/part1/a.out
55cd54029000-55cd5402b000 r-xp 00002000 08:05 2228632
/home/nguyensang/Desktop/OS2021/lab4/part1/a.out
55cd5402b000-55cd5402c000 r--p 00004000 08:05 2228632
/home/nguyensang/Desktop/OS2021/lab4/part1/a.out
55cd5402c000-55cd5402d000 r--p 00004000 08:05 2228632
/home/nguyensang/Desktop/OS2021/lab4/part1/a.out
55cd5402d000-55cd5402e000 rw-p 00005000 08:05 2228632
/home/nguyensang/Desktop/OS2021/lab4/part1/a.out
55cd55f87000-55cd55fa8000 rw-p 00000000 00:00 0                          [heap]
7fafb425d000-7fafb4282000 r--p 00000000 08:05 2755622
/usr/lib/x86_64-linux-gnu/libc-2.31.so
7fafb4282000-7fafb43fa000 r-xp 00025000 08:05 2755622
/usr/lib/x86_64-linux-gnu/libc-2.31.so
7fafb43fa000-7fafb4444000 r--p 0019d000 08:05 2755622
/usr/lib/x86_64-linux-gnu/libc-2.31.so
7fafb4444000-7fafb4445000 ---p 001e7000 08:05 2755622
/usr/lib/x86_64-linux-gnu/libc-2.31.so
7fafb4445000-7fafb4448000 r--p 001e7000 08:05 2755622
/usr/lib/x86_64-linux-gnu/libc-2.31.so
7fafb4448000-7fafb444b000 rw-p 001ea000 08:05 2755622
/usr/lib/x86_64-linux-gnu/libc-2.31.so
7fafb444b000-7fafb4451000 rw-p 00000000 00:00 0
```

```
7fafb4464000-7fafb4465000 r--p 00000000 08:05 2755618
/usr/lib/x86_64-linux-gnu/ld-2.31.so
7fafb4465000-7fafb4488000 r-xp 00001000 08:05 2755618
/usr/lib/x86_64-linux-gnu/ld-2.31.so
7fafb4488000-7fafb4490000 r--p 00024000 08:05 2755618
/usr/lib/x86_64-linux-gnu/ld-2.31.so
7fafb4491000-7fafb4492000 r--p 0002c000 08:05 2755618
/usr/lib/x86_64-linux-gnu/ld-2.31.so
7fafb4492000-7fafb4493000 rw-p 0002d000 08:05 2755618
/usr/lib/x86_64-linux-gnu/ld-2.31.so
7fafb4493000-7fafb4494000 rw-p 00000000 00:00 0
7ffd0285a000-7ffd0287b000 rw-p 00000000 00:00 0                          [stack]
7ffd0291e000-7ffd02922000 r--p 00000000 00:00 0                          [vvar]
7ffd02922000-7ffd02924000 r-xp 00000000 00:00 0                          [vdso]
ffffffffff600000-ffffffffff601000 --xp 00000000 00:00 0
[vsyscall]

[--mem--]

[--root--]
/proc/4029/root ->  /

[--stat--]
pid                     4029
comm                    (a.out)
state                   R
ppid                    1483
pgrp                    4028
session                 4028
tty_nr                  0
tpgid                   -1
flags                   4194368
minflt                  46
cminflt                 0
majflt                  0
cmajflt                 0
utime                   38
stime                   16
cutime                  0
cstime                  0
priority                20
nice                    0
num_threads             1
itrealvalue             0
starttime               22628
vsize                   2564096
rss                     24
rsslim                  18446744073709551615
startcode               94340366110720
endcode                 94340366116229
startstack              140724645894672
kstkesp                 0
kstkeip                 0
signal                  0
blocked                 0
sigignore               1
sigcatch                0
wchan                   0
nswap                   0
cnswap                  0
exit_signal             17
processor               0
rt_priority             0
policy                  0
delayacct_blkio_ticks   0
guest_time              0
cguest_time             0
```

```
start_data              94340366126256
end_data                94340366127648
start_brk               94340399001600
arg_start               140724645896943
arg_end                 140724

[--statm--]
size                    626
resident                24
shared                  0
text                    2
lib                     0
data                    78
```