МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Национальный исследовательский ядерный университет «МИФИ»

(НИЯУ МИФИ)

Институт интеллектуальных кибернетических систем

Кафедра Кибернетики

**Лабораторная работа №2 по курсу**

**«Разработка ПО ОС UNIX»**

**Выполнил студент группы Б15-501:** Огнянович Павел

**Проверил:**  Ктитров С.В.

Москва, 2018

**Задание**

Разработать программу, записывающую данные дочерним процессом в файл, полученный от родительского процесса через наследование дескриптора (имени файла процесс не знает). Значение дескриптора передается через переменную окружения или аргумент. Программу оформить как утилиту командной строки.

**Код программы**

Lab-2.c

#include <fcntl.h>

#include <unistd.h>

#include <sys/stat.h>

#include <stdlib.h>

#include <errno.h>

#include <stdio.h>

int main(int argc, char \*argv[]) {

int carg;

char \*filename;

while((carg=getopt(argc, argv,"f:h"))!=-1) {

switch(carg) {

case 'f':

filename = optarg;

break;

case 'h':

printf("USAGE:\n\t-h help\n\t-f filename\n");

return 0;

break;

case '?':

printf("WRONG ARG!");

break;

}

}

int fd = creat(filename, S\_IRWXU | S\_IRWXG | S\_IRWXO);

printf("PARENT: fd=%d\n",fd);

pid\_t pid;

switch(pid = fork()) {

case -1:

printf("fork ERROR!");

break;

case 0:

printf("CHILD\_PROCESS CASE\n");

execl("lab-2-2.exe", fd, NULL);

break;

default:

printf("DEFAULT CASEn\n");

}

return 0;

}

Lab-2-2.c

#include <stdio.h>

#include <stdlib.h>

#include <fcntl.h>

#include <sys/stat.h>

#include <sys/types.h>

#include <string.h>

int main(int argc, char \*argv[]) {

printf("CHILD CODE EXECUTION\n");

char \*ch = argv[1];

if(ch==NULL) {

printf("COMMAND\_LINE\_ARGUMENT\_ERROR!");

return 1;

}

int fd = atoi(ch);

printf("CHILD: fd=%d\n",fd);

char buff[] = "1234567890";

int size = strlen(buff);

if(write(fd, buff, size)!=size)

printf("WRITE\_ERROR!\n");

close(fd);

return 0;

}