Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

Лабораторная работа №6

за 4 семестр

По дисциплине: «ОСиСП»

Тема: «Средства межпроцессного взаимодействия»

Выполнила:

Студентка 2 курса

Группы ПО-3(1

Давидюк Ю. И.

Выполнила:

Студентка 2 курса

Группы ПО-3(1)

Гаврилюк Р.И.

Проверил:

Давидюк Ю. И

Выполнил:

Студент 2 курса

Группы ПО-4(2)

Кречко К.А.

Проверила:

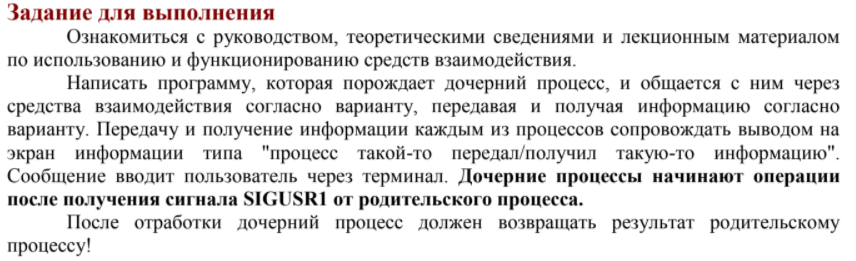
Давидюк Ю.И.

Брест 2021

Лабораторная работа №6

«Средства межпроцессного взаимодействия»

**Вариант – 14(4)**

****

****

**Код:**

#include <stdio.h>

#include <sys/mman.h>

#include <unistd.h>

#include <fcntl.h>

#include <sys/types.h>

#include <stdlib.h>

#include <signal.h>

void\* my\_handler(int nsig){

printf("signal is works!\n");

}

int Factorial(int N){ //фун-я нахождения факториала

if(N < 1)

return 0;

if(N == 1)

return 1;

return N\*Factorial(N - 1);

}

int main(void) {

int fd;

int \*region;

size\_t pagesize = getpagesize(); //получение размера страницы

signal(SIGUSR1, my\_handler);

//создание файла для общего доступа

if((fd = open("myfile", O\_RDWR | O\_CREAT|O\_TRUNC,0666)) < 0){

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

return -1;

}

(void)umask(0);

write (fd,"/0", sizeof(char)); //пустой файл не появится; передаем /0

region=(int\*)mmap(

NULL, // Начинаем с адреса 0 (с начала)

pagesize, // длина в одну страницу

PROT\_READ|PROT\_WRITE, // можно читать и записывать информацию

MAP\_SHARED,

fd, 0);

if (region == MAP\_FAILED) { //проверка на открытие mmap

perror("Could not mmap");

return -1;

}

//Порождаем процесс

printf("родитель процесс PID = %d PPID = %d\n", getpid(), getppid());

pid\_t pid;

printf("Введите число: ");

scanf("%i", &region[0]);

pid = fork();

if(pid == -1){

printf("Ошибкa\n");

exit(-1);

}

if (pid > 0) { //родитель

printf("Parent process\n");

kill(pid, SIGUSR1); //сигнал

sleep(1);

printf("Parent process 2\n");

printf("Result= ");

printf("%i\n", region[0]);

printf("Завершился родительский процесс PID = %d PPID = %d\n", getpid(),getppid());

exit(0);

} else{ //ребёнок

printf("Child process \n");

printf("принятое число= ");

printf("%i\n", region[0]);

region[0] = Factorial(region[0]); //вызов фун-и

printf("факториал= ");

printf("%d\n", region[0]);

if (munmap(&region[0], pagesize) < 0 ) { //обновление файла

perror("Could not munmap");

return -1;

}

printf("Завершился дочерний процесс PID = %d PPID = %d\n", getpid(), getppid());

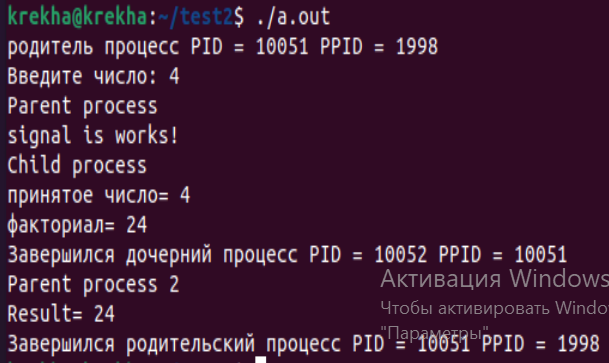
exit(0);

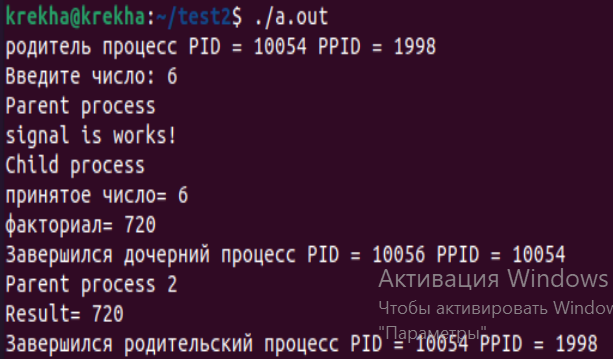
}

return 0;

}

**Результат:**

****



**Вывод:** Я ознакомился со средствами межпроцессорного взаимодействия.