

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(2)
Овсейчук В.И.
Проверил:
Давидюк Ю.И.

Лабораторная работа №6

Вариант 19

19	Общие файлы	Родитель передает потомку пять строк, потомок возвращает строку, в которой есть палиндром
----	-------------	---

Содержимое lab6.c

```
#include <stdio.h>
#include <sys/mman.h>
#include <unistd.h>
#include <fcntl.h>
#include <signal.h>
#include <string.h>
#include <stdlib.h>
int is_palindrome(char* s)
{
    for (int i = 0, j = strlen(s) - 1; i < j; ++i, --j)
        if (s[i] != s[j]) return 0;
    return 1;
}
void handler(int i)
{
    printf("\nparent sent signal to the child process \n\n");
}

int main(void) {
    signal(SIGUSR1, handler);
    int fd; size_t pagesize = getpagesize();
    char **region;
    char * str1 = "hello";
    char * str2 = "lala";
    char * str3 = "reer";
    char * str4 = "lerok";
    char * str5 = "wqwq";

    //create file for shared access
    if ((fd = open("myfile", O_RDWR | O_CREAT | O_TRUNC, 0777)) < 0) {
        printf("Can't open file\n");
        return -1;
    }
    //empty file is not mapping
    write(fd, "\0", sizeof(char));
    region = (char**)mmap(
        NULL, // Map from the start of the null page
        pagesize, // for one page length
        PROT_READ | PROT_WRITE, // to a private block of
        // hardware memory
        MAP_SHARED,
        fd, 0);
    if (region == MAP_FAILED) {
        perror("Could not mmap");
        return -1;
    }
    //write strings
    region[0] = str1;
    region[1] = str2;
    region[2] = str3;
    region[3] = str4;
```

```

region[4] = str5;

//create process
int result = fork();
if (result < 0) {
    printf("Can\'t fork child\n");
    return -1;
}
else if (result > 0) {
    //parent process
    printf("Shared file (in parent): \n");
    for (int i = 0; i < 5; i++) {
        printf("%s \n", region[i]);
    }
    //parent process is stopped, jumping into child process
    kill(result, SIGUSR1);
    sleep(1);

    if (munmap(region, pagesize) < 0)
    {
        perror("Could not munmap");
        return -1;
    }
    printf("parent process exit\n");
}
else {
    //child process
    char ** stringg[5][100];
    for (int i = 0; i < 5; i++) {
        strcpy(stringg[i], region[i]);
    }
    printf("Shared file (in child): \n");
    for (int i = 0; i < 5; i++) {
        printf("%s \n", stringg[i]);
    }
    for (int i = 0; i < 5; i++) {
        if (is_palindrome(stringg[i])) {
            printf("\nWe found palindrome :\n");
            printf("%s\n\n", stringg[i]);
        }
    }
    printf("child process exit\n");
    exit(0);
}
return 0;
}

```

Вывод программы:

```
lera@lera-VirtualBox:~$ ./a.out
Shared file (in parent):
hello
lala
reer
lerok
wqwq

parent sent signal to the child process

Shared file (in child):
hello
lala
reer
lerok
wqwq

We found palindrome :
reer

child process exit
parent process exit
```

Вывод: Изучила средства межпроцессорного взаимодействия.