Пермский филиал федерального государственного автономного образовательного учреждения высшего образования   
«Национальный исследовательский университет   
«Высшая школа экономики»

*Факультет информационных технологий в бизнесе*

Можегова Анна Сергеевна

**Лабораторная работа №8 «Взаимодействие процессов POSIX» по дисциплине**

**«Операционные системы»**

**Вариант №2: процесс, Неименованный канал**

студента образовательной программы «Программная инженерия»

по направлению подготовки *09.03.04 Программная инженерия*

**Руководитель:**

Кузнецов Денис  
 Борисович

Пермь, 2023 год

#include <stdio.h>

#include <stdlib.h>

#include <unistd.h>

#include <signal.h>

#include <string.h>

#include <sys/wait.h>

#define BUFFER\_SIZE 256

int fd[2];

pid\_t child\_pid;

void signal\_handler(int signum) {

if (signum == SIGUSR1) {

printf("Дочерний процесс получил сигнал от родителя\n"); char buffer[BUFFER\_SIZE];

strcpy(buffer, "строка"); write(fd[1], buffer, strlen(buffer) + 1);

printf("Дочерний процесс отправил: %s\n", buffer);

}

}

int main() {

pipe(fd);

child\_pid = fork();

if (child\_pid == -1) { perror("Ошибка при вызове функции fork()"); exit(EXIT\_FAILURE);

}

if (child\_pid == 0) {

close(fd[0]);

signal(SIGUSR1, signal\_handler);

printf("Дочерний процесс ждет \n");

pause();

close(fd[1]);

exit(EXIT\_SUCCESS);

}

else {

close(fd[1]);

printf("Родитель отправляет сигнал \n”);

kill(child\_pid, SIGUSR1);

printf("Родитель ожидает сообщение от дочернего процесса\n");

char buffer[BUFFER\_SIZE];

read(fd[0], buffer, BUFFER\_SIZE);

printf("Родительский процесс получил: %s\n", buffer); close(fd[0]);

}

return 0;

}

Изображение выглядит как текст, снимок экрана, Шрифт, черный

Автоматически созданное описание