

Министерство образования Республики Беларусь
Учреждение образования
“Брестский Государственный технический университет”
Кафедра ИИТ

Лабораторная работа №6
По дисциплине “Операционные системы”
Тема: “Средства межпроцессорного взаимодействия”

Выполнил:
Студент 2 курса
Группы ИИ-21
Кирилович А. А.
Проверил:
Козинский А. А.

Цел работы: изучить средства межпроцессорного взаимодействия.

Ход работы:

Задание:

Написать программу, которая порождает дочерний процесс, и общается с ним через средства взаимодействия согласно варианту (Именованные каналы), передавая и получая информацию согласно варианту (Родитель передает величины катетов прямоугольного треугольника, назад получает величины острых углов.). Передачу и получение информации каждым из процессов сопровождать выводом на экран информации типа "процесс такой-то передал/получил такую-то информацию". Дочерние процессы начинают операции после получения сигнала SIGUSR1 от родительского процесса.

```
#include <stdio.h>
#include <unistd.h>
#include <stdlib.h>
#include <signal.h>
#include <string.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <errno.h>
#include <math.h>

#define MQ_NAME "my_fifo"

int fd_fifo;
char buf[20];

void sig_handler_parent(int signum) {
    printf("Parent : Received a response signal from child \n");
    read(fd_fifo, buf, 20);
    printf("Answer: %s\r\n", buf);
}

void sig_handler_child(int signum) {
    printf("Child : Received a signal from parent \n");
    read(fd_fifo, buf, 20);
    int a, b;
    sscanf(buf, "%d %d", &a, &b);
    float c = sqrt(a*a + b*b);
    float alpha = acos(a/c);
    float beta = acos(b/c);
    sprintf(buf, "%f %f", alpha, beta);
    write(fd_fifo, buf, strlen(buf));
    buf[0] = '\0';
    sleep(1);
    kill(getppid(), SIGUSR1);
}

int main() {
    pid_t pid;

    unlink(MQ_NAME);
    (void)umask(0);

    if ((mkfifo(MQ_NAME, 0666)) != 0) {
        perror("Невозможно создать fifo\n");
        exit(1);
    }
    if ((fd_fifo = open(MQ_NAME, O_RDWR)) == - 1) {
        perror("Невозможно открыть fifo\n");
        exit(1);
    }

    if ((pid = fork())<0){
        printf("Fork Failed\n");
        exit(1);
    }
}
```

```

/* Child Process */
else if(pid == 0) {
    signal(SIGUSR1,sig_handler_child);
    printf("Child: waiting for signal\n");
    pause();
}
/* Parent Process */
else {
    signal(SIGUSR1,sig_handler_parent);
    sleep(1);
    sprintf(buf, "%d %d", 3, 4);
    write(fd_fifo, buf, strlen(buf));
    buf[0] = '\0';
    printf("Parent: sending signal to Child\n");
    kill(pid,SIGUSR1);
    printf("Parent: waiting for response\n");
    pause();
}
return 0;
}

/*Child: waiting for signal
Parent: sending signal to Child
Parent: waiting for response
Child : Received a signal from parent
Parent : Received a response signal from child
Answer: 0.927295 0.643501
*/

```

Вывод: изучил средства межпроцессорного взаимодействия на примере именованных каналов.