Московский Авиационный Институт

(Национальный Исследовательский Университет)

Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

Лабораторная работа №4 по курсу

«Операционные системы»

ФАЙЛОВЫЕ СИСТЕМЫ

Студент: Белоносов Кирилл Алексеевич
Группа: М8О–208Б–22
Вариант: 13
Преподаватель: Соколов Андрей Алексеевич
Оценка: _____
Дата: _____
Подпись: _____

Москва, 2022.

Постановка задачи

Цель работы

Приобретение практических навыков в:

- Освоение принципов работы с файловыми системами
- Обеспечение обмена данных между процессами посредством технологии «File mapping»

Задание

Составить и отладить программу на языке Си, осуществляющую работу с процессами и взаимодействие между ними в одной из двух операционных систем. В результате работы программа (основной процесс) должен создать для решение задачи один или несколько дочерних процессов. Взаимодействие между процессами осуществляется через системные сигналы/события и/или через отображаемые файлы (memory-mapped files).

Необходимо обрабатывать системные ошибки, которые могут возникнуть в результате работы.

Общие сведения о программе

Программа состоит из одного файла main.c. В ходе работы программы создается 2 дочерних процесса, первый переводит все символы в нижний регистр, второй заменяет все пробельные символы на нижние подчеркивания. Межпроцессорное взаимодействие осуществляется благодаря сигналам.

Системные вызовы:

1. **kill()** - посылает сигнал процессу
2. **waitpid()** - дождаться завершения процесса
3. **pause()** - дождаться сигнала
4. **signal()** - установить обработчик сигнала
5. **mmap()** - отразить файл в память

Общий метод и алгоритм решения.

Так как большинство функционала для работы с лабораторной работой было реализовано еще во второй лабораторной работе, то нужно лишь перестроить способ межпроцессорного взаимодействия и использовать mmap

Для реализации поставленной задачи необходимо:

1. Изучить принцип работы mmap
2. Познакомиться с принципом работы сигналов
3. Организовать межпроцессорное взаимодействие с использованием сигналов
4. Провести тестирование

Основные файлы программы

main.c:

```
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
#include <sys/wait.h>
#include <string.h>
#include <sys/mman.h>
#include <signal.h>
void sig_handler(int signum){
}
void sig_handler_1(int signum){
   kill(getppid(), SIGUSR1);
}
void sig_handler_2(int signum){
   kill(getppid(), SIGUSR1);
}
int main (void) {
   int pid1, pid2;
   char * ptr = mmap(NULL, 1, PROT_READ|PROT_WRITE, MAP_SHARED|
MAP_ANONYMOUS, 0, 0);
```

```
if(ptr == MAP_FAILED){
    perror("Not mmap");
}
if((pid1 = fork()) == 0) {
    int loop = 1;
    signal(SIGINT, SIG_DFL);
    signal(SIGUSR1, sig_handler_1);
    while (loop)
    {
        pause();
        if(ptr[0] >= 'A' && ptr[0] <= 'Z') {
            ptr[0] = ptr[0] + 32;
        }
    }
}
if(pid1 > 0 && (pid2 = fork()) == 0) {
    int loop = 1;
    signal(SIGINT, SIG_DFL);
    signal(SIGUSR1, sig_handler_2);
    while (loop)
    {
        pause();
        if(ptr[0] == ' ') {
            ptr[0] = '_';
        }
    }
}
if(pid1 == -1 || pid2 == -1) {
    perror("don't create child process");
    exit(EXIT_FAILURE);
}
if(pid1 == -1) {
    perror("don't create child1 process");
```

```
        exit(EXIT_FAILURE);
    }
    if(pid2 == -1) {
        perror("don't create child2 process");
        exit(EXIT_FAILURE);
    }
    if(pid1 != 0 && pid2 != 0) {
        signal(SIGUSR1, sig_handler);
        char c;
        int err;
        while(err = read(0, &c, 1) > 0) {
            ptr[0] = c;
            kill(pid1, SIGUSR1);
            kill(pid2, SIGUSR1);
            pause();
            printf("%c", ptr[0]);
        }
        kill(pid1, SIGINT);
        kill(pid1, SIGINT);
        waitpid(-1, NULL, 0);
        int errump = munmap(ptr, 1);
        if(errump != 0){
            perror("don't create child2 process");
            exit(EXIT_FAILURE);
        }
    }
    return 0;
}
```

Пример работы

```
kirill@kirill:~/localProjects/OSlabs/lab4/build/src$
/home/kirill/localProjects/OSlabs/lab4/build/src/main

THE QUICK BROWN FOX
```

the_quick_brown_fox

HELLO WORLD

hello_world

It My lab

it_my_lab

Вывод

В данной лабораторной я познакомился с механизмами работы с файловыми системы, в частности с системным вызовом mmap. Также я реализовал межпроцессорное взаимодействие с помощью сигналов. Благодаря данной лабораторной работе я понял, что существует несколько способов организации общения процессов, выбор которого зависит от конкретной задачи.