Московский Авиационный Институт

(Национальный Исследовательский Университет)

Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

**Лабораторная работа №4 по курсу**

**«Операционные системы»**

Студент: Уваров А.П.

Группа: М8О–206Б–19

Вариант: 19

Преподаватель: Соколов Андрей Алексеевич

Оценка: \_\_\_\_\_\_\_\_\_\_

Дата: \_\_\_\_\_\_\_\_\_\_

Подпись: \_\_\_\_\_\_\_\_\_\_

Москва, 2020.

**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.

Вариант 13: Родитель из потока ввода передает строки двум детям в соотношении 80/20, они из строк удаляют гласные и вписывают результат в соответствующие файлы.

**Общие сведения о программе**

В программе используются следующие системные вызовы:

1. **mmap** – создает отображение файла в память.
2. **munmap** – снимает отображение.
3. **open** – открывает файл.
4. **close –** закрывает файл.
5. **read** – чтение из файла в буфер.
6. **write** – запись из буфера в файл.
7. **sleep** – переход в режим ожидания на указанное количество секунд.

**Общий метод и алгоритм решения**.

Для реализации поставленной задачи необходимо:

1. Изучить работу с отображением файла в память(mmap и munmap).
2. Изучить работу с процессами(fork).
3. Создать 2 дочерних и 1 родительский процесс.
4. В каждом процессе отобразить файл в память, преобразовать в соответствии с вариантом и снять отображение(mmap, munmap).

**Основные файлы программы**

**var\_19\_main.c:**

**#include <unistd.h>**

**#include <stdlib.h>**

**#include <stdio.h>**

**#include <sys/types.h>**

**#include <sys/stat.h>**

**#include <fcntl.h>**

**#include <sys/mman.h>**

**#include <time.h>**

**#include <string.h>**

**#include "lab\_4.h"**

**const int SIZE\_OF\_CHARS = 200;**

**int main(){**

**srand(time(0));**

**char path1[200] = "",path2[200] = "";**

**char t[200] = "Введи в следующий двух строках - пути к файлам, куда я буду отправлять выходные данные ^\_^\n";**

**write(0,t,sizeof(char) \* 200);**

**read(0,path1,sizeof(char)\*200);**

**int path1Len = strlen(path1);**

**path1[path1Len-1] = '\0';**

**if(!strcmp(path1, "pip1") || !strcmp(path1, "ex1") ||**

**!strcmp(path1, "pip2") || !strcmp(path1, "ex2")){**

**char k[200] = "Такое название файлов недопустимо, попробуйте снова!\n";**

**write(0,k,sizeof(char) \* 200);**

**return 1;**

**}**

**read(0,path2,sizeof(char)\*200);**

**int path2Len = strlen(path2);**

**path2[path2Len-1] = '\0';**

**if(!strcmp(path2, "pip2") || !strcmp(path2, "ex2") ||**

**!strcmp(path2, "pip1") || !strcmp(path2, "ex1")){**

**char k[200] = "Такое название файлов недопустимо, попробуйте снова!\n";**

**write(0,k,sizeof(char) \* 200);**

**return 1;**

**}**

**int ft1 = open(path1,O\_WRONLY | O\_CREAT | O\_APPEND, S\_IRWXU);**

**int ft2 = open(path2,O\_WRONLY | O\_CREAT | O\_APPEND, S\_IRWXU);**

**if(ft1 == -1 || ft2 == -1){**

**char k[200] = "С файлами произошла неизвестная ошибка!\n";**

**write(0,k,sizeof(char) \* 200);**

**return 1;**

**}**

**// int fd[2];**

**// int fd2[2];**

**int pip1int = open("pip1", O\_CREAT | O\_RDWR, S\_IRWXU);**

**char\* mm1 = (char\*)mmap(NULL, SIZE\_OF\_CHARS, PROT\_READ | PROT\_WRITE, MAP\_SHARED, pip1int, 0);**

**ftruncate(pip1int, SIZE\_OF\_CHARS);**

**int ex1int = open("ex1", O\_CREAT | O\_RDWR, S\_IRWXU);**

**char\* mme1 = (char\*)mmap(NULL, 1, PROT\_READ | PROT\_WRITE, MAP\_SHARED, ex1int, 0);**

**ftruncate(ex1int, 1);**

**int pip2int = open("pip2", O\_CREAT | O\_RDWR, S\_IRWXU);**

**char\* mm2 = (char\*)mmap(NULL, SIZE\_OF\_CHARS, PROT\_READ | PROT\_WRITE, MAP\_SHARED, pip2int, 0);**

**ftruncate(pip2int, SIZE\_OF\_CHARS);**

**int ex2int = open("ex2", O\_CREAT | O\_RDWR, S\_IRWXU);**

**char\* mme2 = (char\*)mmap(NULL, sizeof(char\*), PROT\_READ | PROT\_WRITE, MAP\_SHARED, ex2int, 0);**

**ftruncate(ex2int, 1);**

**if(mm1 == MAP\_FAILED || mm2 == MAP\_FAILED ||**

**mme1 == MAP\_FAILED || mme2 == MAP\_FAILED)**

**{**

**char k[200] = "При создании mmap произошла ошибка!\n";**

**write(0,k,sizeof(char) \* 200);**

**return 1;**

**}**

**int id = fork();**

**int id2 = fork();**

**char in[SIZE\_OF\_CHARS];**

**if(id != 0 && id2 != 0){**

**char a[50] = "Введите свои строки\n";**

**write(0,a,50\*sizeof(char));**

**while((scanf("%s",in)) == 1){**

**if(rand() % 100 < 80){**

**char a[50] = "Первый файл\n";**

**write(0,a,50\*sizeof(char));**

**mme1[0] = '+';**

**strcpy(mm1, in);**

**}**

**else{**

**char a[50] = "Второй файл\n";**

**write(0,a,50\*sizeof(char));**

**mme2[0] = '+';**

**strcpy(mm2, in);**

**}**

**}**

**mme2[0] = '-';**

**mme1[0] = '-';**

**}**

**else if(id2 == 0 && id != 0){**

**while(1){**

**while(mme2[0] != '+' && mme2[0] != '-') {}**

**if(mme2[0] == '-')**

**break;**

**mme2[0] = '\_';**

**write(ft2, Task(mm2), SIZE\_OF\_CHARS);**

**printf("[2] Успешно записано! [2]\n");**

**}**

**}**

**else if(id == 0 && id2 != 0){**

**while(1){**

**while(mme1[0] != '+' && mme1[0] != '-') {}**

**if(mme1[0] == '-')**

**break;**

**//printf("[1] I catch PLUS\n");**

**mme1[0] = '\_';**

**write(ft1, Task(mm1), SIZE\_OF\_CHARS);**

**printf("[1] Успешно записано! [1]\n");**

**}**

**}**

**close(ft1);**

**close(ft2);**

**remove("ex1");**

**remove("ex2");**

**remove("pip1");**

**remove("pip2");**

**return 0;**

**}**

**Пример работы**

**andrey@andrey-VivoBook:~ /Рабочий стол/OC/OChelp/4/own$** ./a.out

Введи в следующий двух строках - пути к файлам, куда я буду отправлять выходные данные ^\_^

res

res2

Введите свои строки

qwertyui

Первый файл

[1] Успешно записано! [1]

cvbnm

Второй файл

[2] Успешно записано! [2]

**andrey@andrey-VivoBook:~ /Рабочий стол/OC/OChelp/4/own$** cat res

qwrt

**andrey@andrey-VivoBook:~ /Рабочий стол/OC/OChelp/4/own$** cat res2

cvbnm

**Вывод**

Mmap еще один способ взаимодействия между процессами, такой подход в отличие от pipe ускоряет работу программы, засчет того, что нет вызовов read, write, помимо всего прочего мы возвращаем void \*, который в итоге можем конвертировать под свои типы данных.