

Міністерство освіти і науки України
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій

Звіт
про виконання лабораторної роботи №7
“Реалізація міжпроцесової взаємодії на основі інтерфейсу файлової системи”

Виконав:
студент групи ФЕП-13
Андріан Карсанашвілі
Прийняв:
доц. Бойко Я.В.

Львів-2021

Мета: реалізувати міжпроцесову взаємодію на основі інтерфейсу файлової системи.

Операційна система: Ubuntu 18.04.

Завдання 1

Код:

```
#include <pthread.h>
#include <unistd.h>
#include <sys/mman.h>
#include <fcntl.h>

void* server(void* par) {
    int fdl;
    int* map;

    fdl = open("tmp", O_RDWR | O_CREAT, 0644);
    lseek(fdl, sizeof(int), SEEK_SET);
    write(fdl, "", 1);
    map = (int*)mmap(0, sizeof(int), PROT_WRITE | PROT_READ, MAP_SHARED, fdl, 0);
    close(fdl);

    map[0] = 5;
    while(1) {
        printf("Server temperature is : %i\n", map[0]);
        if (map[0] == 0)
            break;
        usleep(200);
        map[0]--;
    }
}

int index = 0;
pthread_mutex_t i = PTHREAD_MUTEX_INITIALIZER;
int getIndex(){
    index++;
    return index;
}
```

```

void* client(void* par) {
    usleep(200);
    int fdl;
    int* map;

    pthread_mutex_lock(&i);
    int index = getIndex();
    pthread_mutex_unlock(&i);

    fdl = open("./tmp", O_RDONLY);
    map = (int*)mmap(0, sizeof(int), PROT_READ, MAP_SHARED, fdl, 0);
    close(fdl);

    while(1) {
        printf("client %i temperature is : %i\n", index, map[0]);
        if (map[0] == 0)
            break;
        usleep(200);
    }
}

int main() {
    int clientCount = 3;
    pthread_t serverThread;
    pthread_t clientThreads[clientCount];

    pthread_create(&serverThread, NULL, server, NULL);
    for (int i = 0; i < clientCount; ++i)
        pthread_create(&clientThreads[i], NULL, client, NULL);

    for (int i = 0; i < clientCount; ++i)
        pthread_join(clientThreads[i], NULL);
    pthread_join(serverThread, NULL);

    return 0;
}

```

У даній програмі створено один потім-менеджер та три потоки-клієнти. Потім-менеджер задає значення температури від 5 до 1, а кожен з трьох потоків-клієнтів відображає ці зміни. Дійшовши до нуля, робота завершується.

```

Server set yemperature 5
Client 5
Server set yemperature 4
Client 4
Server set yemperature 3
Client 3
Server set yemperature 2
Client 2
Server set yemperature 1
Client 1

```

Завдання 2

Код:

```
void* server(void* par) {

    mkfifo("serverFIFO", 0644);

    int file;
    char fileName[100], fileEntry[10000];
    int length;

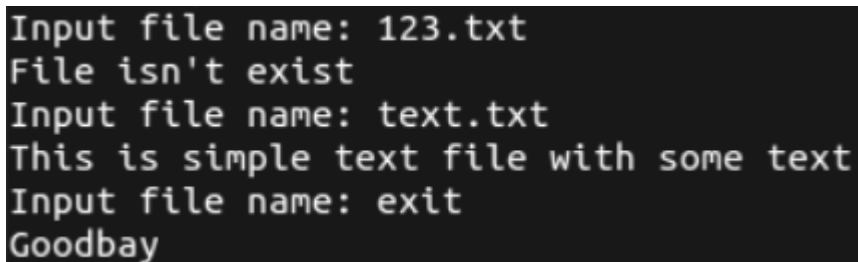
    while(1) {

        file = open("serverFIFO", O_RDONLY);
        length = read(file, fileName, sizeof(fileName));
        close(file);

        if (strcmp(fileName, "exit") == 0)
            break;

        file = open(fileName, O_RDONLY);
        length = read(file, fileEntry, sizeof(fileEntry));
        close(file);

        if (length > 0) {
            file = open("clientFIFO", O_WRONLY);
            write(file, fileEntry, sizeof(fileEntry));
            close(file);
        } else {
            file = open("clientFIFO", O_WRONLY);
            char errorStr[] = "Errorr while try to open file\nMaybe file doesn't exist!";
            write(file, errorStr, sizeof(errorStr));
            close(file);
        }
    }
}
```



```
Input file name: 123.txt
File isn't exist
Input file name: text.txt
This is simple text file with some text
Input file name: exit
Goodbay
```

Створено два потоки: сервер та клієнт. Ми маємо набір з двох файлів: file1 та file2. При вводі назви файлу, сервер зчитує вміст файлу та виводить його на екран. При вводі назви неіснуючого файлу видається помилка. При вводі слова exit робота завершається.

Висновок: Вивчив та застосував механізми міжпроцесової взаємодії на основі інтерфейсу файлової системи.