

Лабораторная работа №14

Операционные системы

Сабралиева Марворид Нуралиевна

Содержание

| | | |
|---|--------------------------------|----|
| 1 | Цель работы | 5 |
| 2 | Выполнение лабораторной работы | 6 |
| | Список литературы | 10 |

Список иллюстраций

| | | |
|-----|-------------------------|---|
| 2.1 | Создаем файлы | 6 |
| 2.2 | файл common.h | 7 |
| 2.3 | файл server.c | 7 |
| 2.4 | файл Makefile | 8 |
| 2.5 | make all | 8 |
| 2.6 | make all | 9 |

Список таблиц

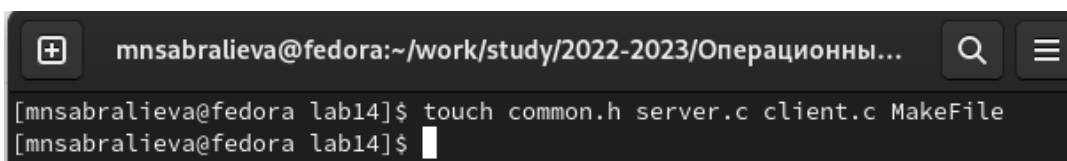
1 Цель работы

Приобретение практических навыков работы с именованными каналами.

2 Выполнение лабораторной работы

Изучив приведённые в тексте программы `server.c` и `client.c`. Взяв данные примеры за образец, мы написали аналогичные программы, внося следующие изменения: 1. Работает не 1 клиент, а несколько (например, два). 2. Клиенты передают текущее время с некоторой периодичностью (например, раз в пять секунд). Используем функцию `sleep()` для приостановки работы клиента. 3. Сервер работает не бесконечно, а прекращает работу через некоторое время (например, 30 сек). Используем функцию `clock()` для определения времени работы сервера

1. Создадим файлы (рис. 2.1).

A screenshot of a terminal window. The title bar shows the user 'mnsabralieva@fedora' and the current directory '~/work/study/2022-2023/Операционны...'. The terminal contains two lines of text: the first line shows the command 'touch common.h server.c client.c MakeFile' being executed, and the second line shows the prompt '[mnsabralieva@fedora lab14]\$' with a cursor, indicating the command has been entered.

```
[mnsabralieva@fedora lab14]$ touch common.h server.c client.c MakeFile
[mnsabralieva@fedora lab14]$
```

Рис. 2.1: Создаем файлы

2. Напишем преобразованные программы в файлы (рис. 2.2).



```
1 #ifndef __COMMON_H__
2 #define __COMMON_H__
3
4 #include <stdio.h>
5 #include <stdlib.h>
6 #include <string.h>
7 #include <errno.h>
8 #include <sys/types.h>
9 #include <sys/stat.h>
10 #include <fcntl.h>
11 #include <unistd.h>
12 #include <time.h>
```

Рис. 2.2: файл common.h



```
1 #include "common.h"
2
3 int main()
4 {
5     int readfd; /* дескриптор для чтения из FIFO */
6     int n;
7     char buff[MAX_BUFF]; /* буфер для чтения данных из FIFO */
8
9     /* баннер */
10    printf("FIFO Server...\n");
11
12    if(mknod(FIFO_NAME, S_IFIFO | 0666, 0) < 0)
13    {
14        fprintf(stderr, "%s: Невозможно создать FIFO (%s)\n", __FILE__,
15            strerror(errno));
16        exit(-1);
17    }
```

Рис. 2.3: файл server.c



Рис. 2.4: файл Makefile

3. Скомпилируем файлы и проверим что вышло (рис. 2.5).

```
./Остановка.
[mnsabralieva@fedora lab14]$ make all
gcc server.c -o server
gcc client.c -o client
[mnsabralieva@fedora lab14]$
```

Рис. 2.5: make all

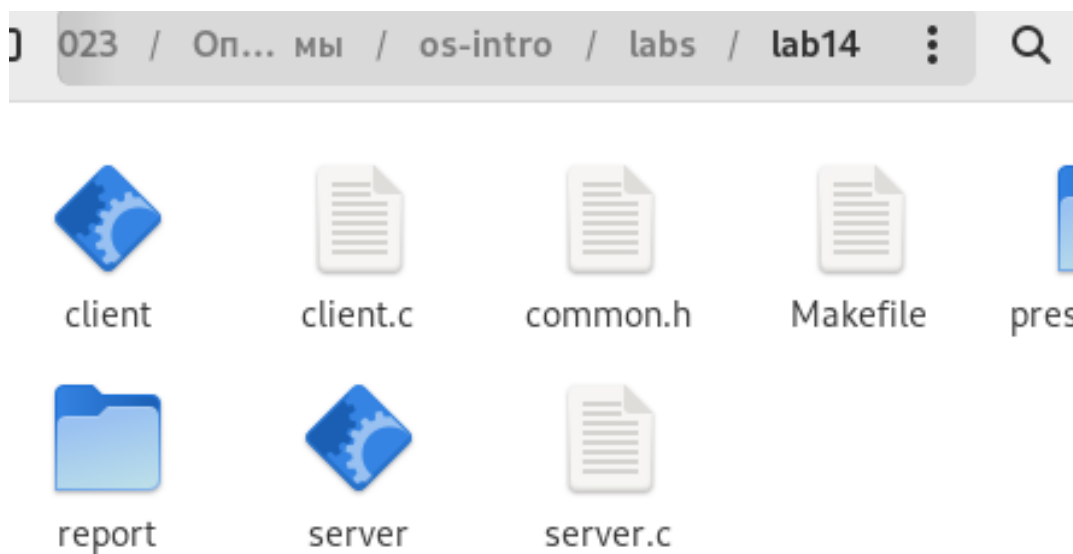


Рис. 2.6: make all

4. Проверим работу программ (рис. ??).

```
[mnsabralieva@fedora lab14]$ ./server
FIFO Server...
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
[mnsabralieva@fedora lab14]$

[mnsabralieva@fedora lab14]$ ./client
FIFO Client...
[mnsabralieva@fedora lab14]$

[mnsabralieva@fedora lab14]$ ./client
FIFO Client...
[mnsabralieva@fedora lab14]$
```

ВЫ-
ВОДЫ

Мы приобрели практические навыки работы с именованными каналами

Список литературы