

Лабораторная работа №14

Именованные каналы

Ежова А. М.

10 мая 2023

Российский университет дружбы народов, Москва, Россия

Факультет физико-математических и естественных наук

Информация

- Ежова Алиса Михайловна
- студентка НБИбд-01-22
- Российский университет дружбы народов
- <https://github.com/AlisaEzhova>

Вводная часть

- Приобретение практических навыков работы с именованными каналами.

- Изучите приведённые в тексте программы `server.c` и `client.c`. Взяв данные примеры за образец, напишите 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). Используйте функцию `sleep()` для приостановки работы клиента.
 3. Сервер работает не бесконечно, а прекращает работу через некоторое время (например, 30 сек). Используйте функцию `clock()` для определения времени работы сервера. Что будет в случае, если сервер завершит работу, не закрыв канал?

- Процессор **pandoc** для входного формата Markdown
- Результирующие форматы
 - pdf
 - html
- Автоматизация процесса создания: **Makefile**

Создание презентации

- Pandoc: преобразователь текстовых файлов
- Сайт: <https://pandoc.org/>
- Репозиторий: <https://github.com/jgm/pandoc>

- Использование LaTeX
- Пакет для презентации: beamer
- Тема оформления: **metropolis**

```
slide_level: 2  
aspectratio: 169  
section-titles: true  
theme: metropolis
```

- Используется фреймворк `reveal.js`
- Используется тема `beige`

- Тема задаётся в файле **Makefile**

```
REVEALJS_THEME = beige
```

Результаты

- Полученный **pdf**-файл можно демонстрировать в любой программе просмотра **pdf**
- Полученный **html**-файл содержит в себе все ресурсы: изображения, **css**, скрипты

1) Я создала необходимые файлы с помощью команды `touch` и открыла редактор для их редактирования:

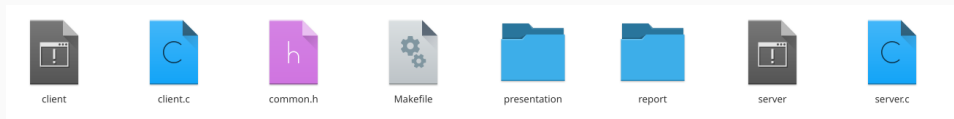
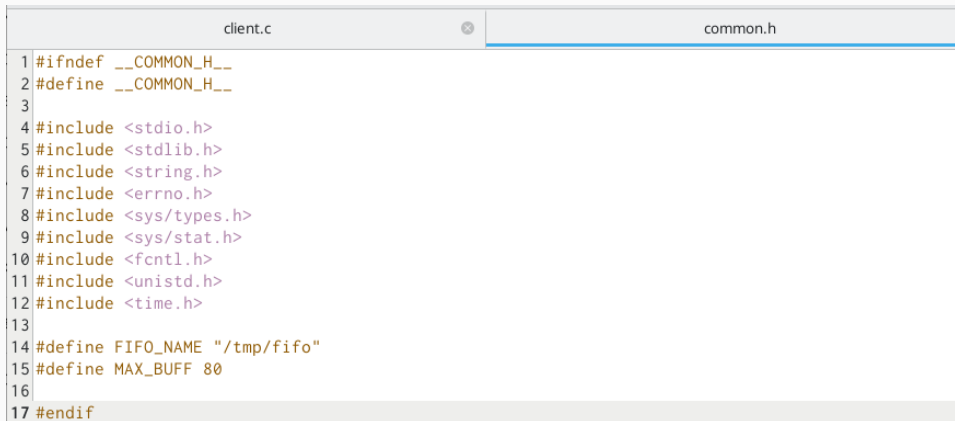


Рис. 1: Созданные файлы

- 2) Далее я изменила коды программ, представленных в тексте лабораторной работы. В файл `common.h` добавила стандартные заголовочные файлы `unistd.h` и `time.h`, необходимые для работы кодов других файлов. `Common.h` предназначен для заголовочных файлов, чтобы в остальных программах их не прописывать каждый раз.



The image shows a code editor with two tabs: `client.c` and `common.h`. The `common.h` tab is active, displaying the following C code:

```
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>
13
14 #define FIFO_NAME "/tmp/fifo"
15 #define MAX_BUFF 80
16
17 #endif
```

Рис. 2: Измененный файл `common.h`

- 3) В файл server.c добавила цикл while для контроля за временем работы сервера. Разница между текущим временем time(NULL) и временем начала работы clock_t start=time(NULL) (инициализация до цикла) не должна превышать 30 секунд:

```
1 #include "common.h"
2
3 int main() {
4     int writefd;
5     int msglen;
6
7     printf("FIFO Client...\n");
8
9     for(int i=0; i<4; i++)
10     {
11
12         if((writefd = open(FIFO_NAME, O_WRONLY)) < 0)
13         {
14             fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n", __FILE__, strerror(errno));
15             exit(-1);
16         }
17
18         long int ttime = time(NULL);
19         char* text = ctime(&ttime);
```

Рис. 3: Измененный файл server.c

- 4) В файл client.c добавила цикл, который отвечает за количество сообщений о текущем времени (4 сообщения), которое получается в результате выполнения команд, и команду sleep(5) для приостановки работы клиента на 5 секунд:

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

Рис. 4: Измененный файл client.c

5) Makefile (файл для сборки) не изменяла:

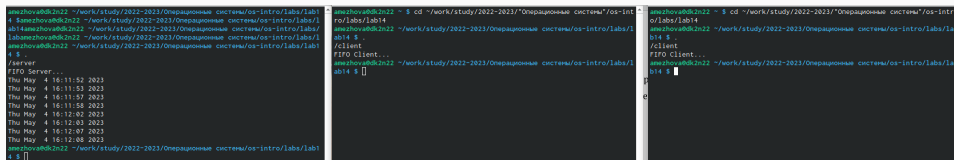
```
all: server client

server: server.c common.h
    gcc server.c -o server

client: client.c common.h
    gcc client.c -o client

clean:
    -rm server client *.o
```

- 6) После написания кодов, я, используя команду «make all», скомпилировала необходимые файлы. Далее я проверила работу написанного кода. Открыла 3 консоли (терминала) и запустила: в первом терминале – «./server», в остальных двух – «./client». В результате каждый терминал-клиент вывел по 4 сообщения. Спустя 30 секунд работа сервера была прекращена. Программа работает корректно:



The image shows three terminal windows side-by-side. The left window runs the server, the middle and right windows run clients. The server prints timestamps and the clients print their IDs.

```
anezhova@dk2n22 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab1
4 $ ./server
FIFO Server...
Thu May 4 16:11:52 2023
Thu May 4 16:11:53 2023
Thu May 4 16:11:57 2023
Thu May 4 16:11:58 2023
Thu May 4 16:12:02 2023
Thu May 4 16:12:03 2023
Thu May 4 16:12:07 2023
Thu May 4 16:12:08 2023
anezhova@dk2n22 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab1
4 $
```

```
anezhova@dk2n22 ~ $ cd ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14
anezhova@dk2n22 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14 $ ./client
FIFO Client...
anezhova@dk2n22 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14 $
```

```
anezhova@dk2n22 ~ $ cd ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14
anezhova@dk2n22 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14 $ ./client
FIFO Client...
anezhova@dk2n22 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14 $
```

Рис. 6: Проверка

- В ходе выполнения Лабораторной работа №14, я приобрела практические навыки работы с именованными каналами.