

# Именованные каналы

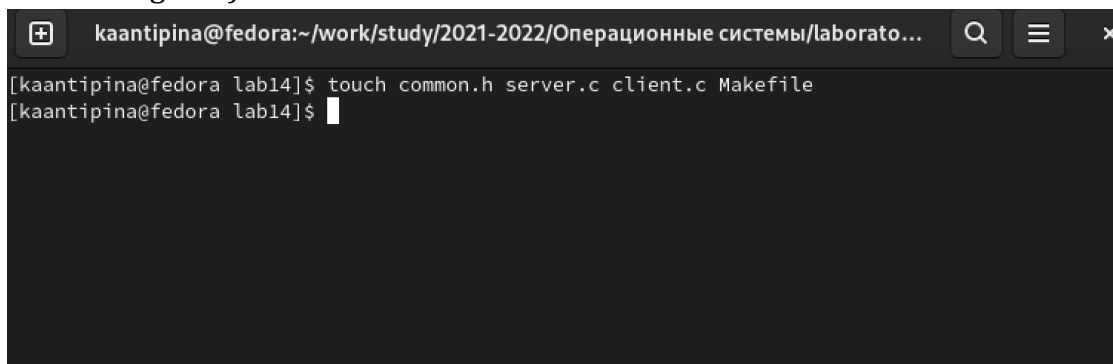
4 июня, 2022, Москва, Россия

## Цель работы

Приобретение практических навыков работы с именованными каналами.

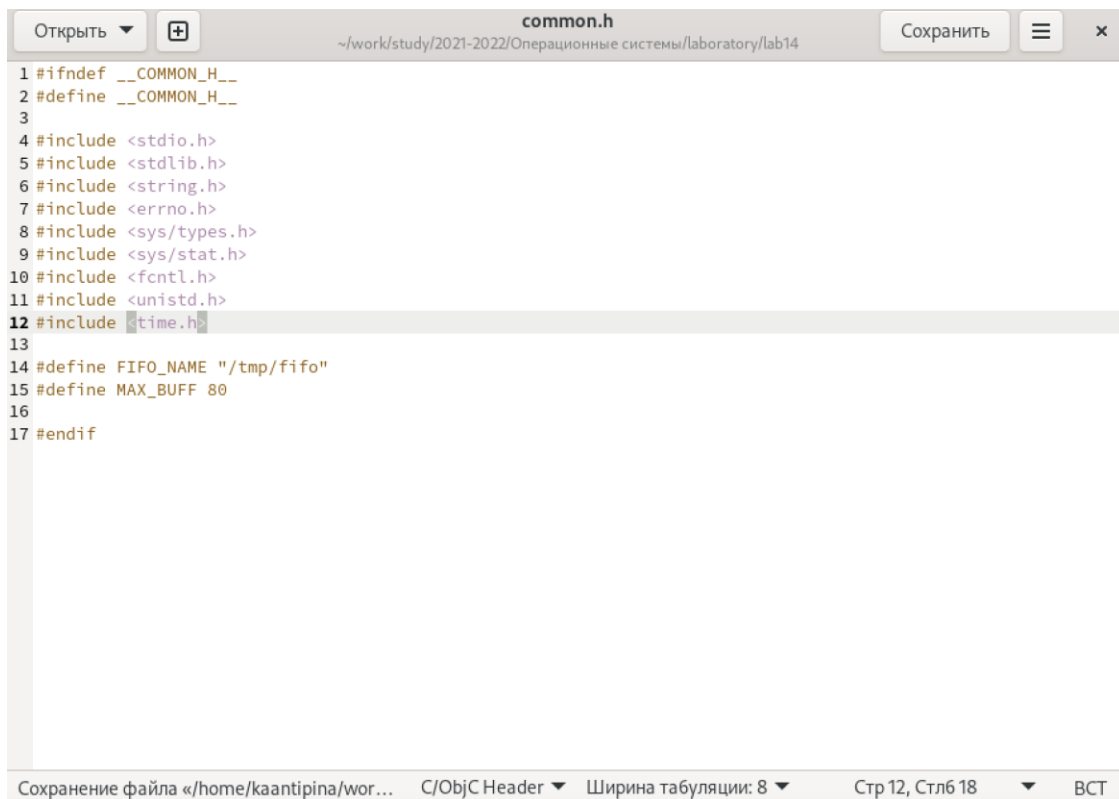
## Ход работы

1. Создаю необходимые файлы с помощью команды «touch common.h server.c client.c Makefile» и открываю редактор emacs для их редактирования.(рис. - @fig:001)

A screenshot of a terminal window. The title bar shows the user 'kaantipina' at 'fedora' in the directory '~/work/study/2021-2022/Операционные системы/laborato...'. The terminal content shows two lines: '[kaantipina@fedora lab14]\$ touch common.h server.c client.c Makefile' and '[kaantipina@fedora lab14]\$' followed by a cursor. The terminal has a dark background and standard Linux window controls.

*Создаю необходимые файлы*

2. Далее я изменю коды программ, представленных в тексте лабораторной работы. В файл common.h добавлю 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(рис. -@fig:002)



```
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
```

Сохранение файла «/home/kaantipina/wor... C/ObjC Header ▾ Ширина табуляции: 8 ▾ Стр 12, Стлб 18 ▾ ВСТ

### Изменяю *common.h*

В файл *server.c* добавлю 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 (рис. - @fig:003) (рис. -@fig:004)

```
Открыть + *server.c ~/work/study/2021-2022/Операционные системы/laboratory/lab14 Сохранить ☰ x
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
16    if((readfd = open(FIFO_NAME, O_RDONLY)) < 0)
17    {
18        fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n", __FILE__, strerror(errno));
19        exit (-2);
20    }
21
22    clock_t start = time(NULL);
23
24    while(time(NULL)-start < 30)
25    {
26        while((n = read(readfd, buff, MAX_BUFF)) > 0)
27        {
28            if(write(1, buff, n) != n)
29            {
30                fprintf(stderr, "%s: Ошибка вывода (%s)\n", __FILE__, strerror(errno));
31                exit(-3);
32            }
33        }
34    }
35
36    close(readfd);
```

### Изменяю server.c

```
34     }
35
36     close(readfd);
37
38     if(unlink(FIFO_NAME) < 0)
39     {
40         fprintf(stderr, "%s: Невозможно удалить FIFO (%s)\n", __FILE__, strerror(errno));
41         exit(-4);
42     }
43
44     exit(0);
45 }
46
47
48
49
50
51
52
53
54
```

### Изменяю server.c

В файл client.c добавлю 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.(рис. -@fig:005)

```
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);
20
21         msglen = strlen(text);
22         if(write(writefd, text, msglen) != msglen)
23         {
24             fprintf(stderr, "%s: Ошибка записи в FIFO (%s)\n", __FILE__, strerror(errno));
25             exit(-2);
26         }
27
28         sleep (5);
29     }
30
31     close(writefd);
32
33     exit(0);
34 }
35
```

С ▾ Ширина табуляции: 8 ▾ Стр 18, Стлб 17 ▾ ВСТ

*Изменяю client.c*

Makefile (файл для сборки) не меняю (рис. -@fig:006)



The screenshot shows a text editor window titled "Makefile" with a path of "~/work/study/2021-2022/Операционные системы/laboratory/lab14". The editor contains the following Makefile content:

```
1 all: server client
2
3 server: server.c common.h
4     gcc server.c -o server
5
6 client: client.c common.h
7     gcc client.c -o client
8
9 clean:
10     -rm server client *.o
```

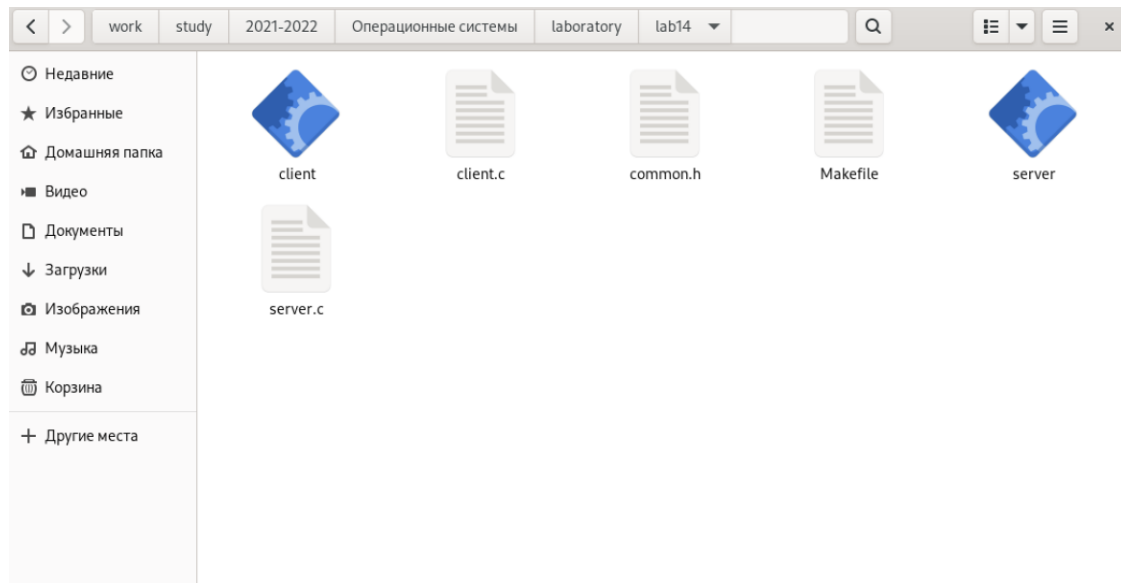
The status bar at the bottom indicates "Makefile", "Ширина табуляции: 8", "Стр 10, Стлб 9", and "ВСТ".

### *Makefile*

3. После написания кодов, я, используя команду «make all», скомпилирую необходимые файлы (рис. -@fig:007)(рис. -@fig:008)

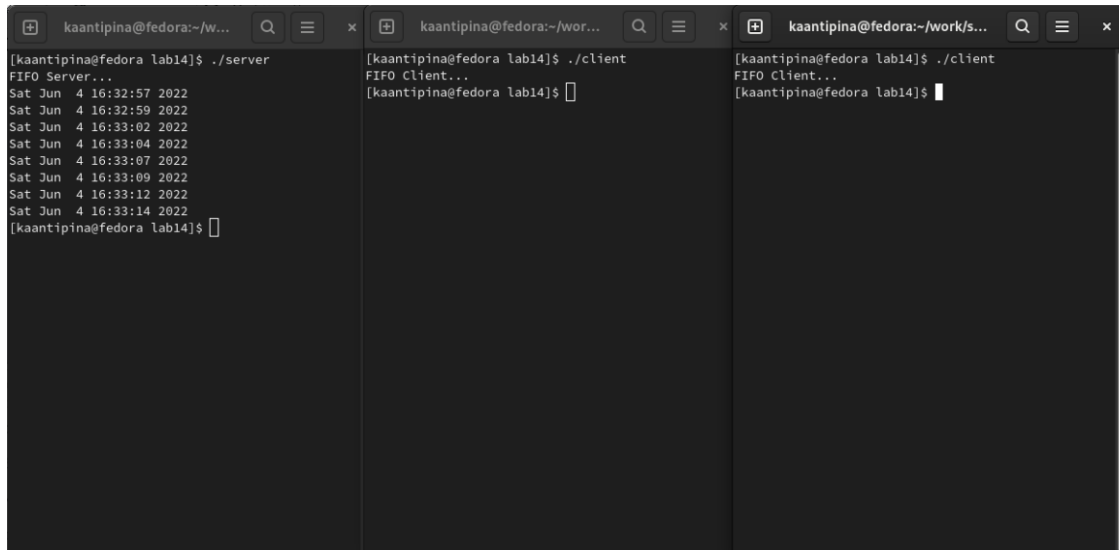
```
kaantipina@fedora:~/work/study/2021-2022/Операционные системы/laborato...  
[kaantipina@fedora lab14]$ touch common.h server.c client.c Makefile  
[kaantipina@fedora lab14]$ make all  
gcc server.c -o server  
gcc client.c -o client  
[kaantipina@fedora lab14]$
```

### Компилирую файлы



### Компилирую файлы

Далее я проверю работу написанного кода. Открою 3 консоли (терминала) и запускаю: 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.(рис. -@fig:009)



```
[kaantipina@fedora lab14]$ ./server
FIFO Server...
Sat Jun 4 16:32:57 2022
Sat Jun 4 16:32:59 2022
Sat Jun 4 16:33:02 2022
Sat Jun 4 16:33:04 2022
Sat Jun 4 16:33:07 2022
Sat Jun 4 16:33:09 2022
Sat Jun 4 16:33:12 2022
Sat Jun 4 16:33:14 2022
[kaantipina@fedora lab14]$

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

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

### *Проверяю работу программ*

Также я проверю длительность работы сервера, введя команду «./server» в одном терминале. Он завершил свою работу через 30 секунд. Если сервер завершит свою работу, не закрыв канал, то, когда мы будем запускать этот сервер снова.(рис. - @fig:010)



```
[kaantipina@fedora lab14]$ ./server
FIFO Server...

./server
clock
```

### *Проверяю длительность работы сервера*

## **Вывод**

В ходе выполнения лабораторной работы № 14 я приобрела практические навыки работы с именованными каналами.