

# Лабораторная работа №14

Именованные каналы

---

Медникова Е. М.

13 мая 2023

Российский университет дружбы народов, Москва, Россия

Факультет физико-математических и естественных наук

## Информация

---

- Медникова Екатерина Михайловна
- студентка направления бакалавриата 01.03.00 Математика и механика
- Российский университет дружбы народов
- 1132226549@rudn.ru

## Цель работы

---

Приобретение практических навыков работы с именованными каналами.

## Задание

---

Изучить приведённые в тексте программы `server.c` и `client.c`. Взяв данные примеры за образец, написать аналогичные программы, внося следующие изменения:

Работает не 1 клиент, а несколько (например, два).

Клиенты передают текущее время с некоторой периодичностью (например, раз в пять секунд). Использовать функцию `sleep()` для приостановки работы клиента.

Сервер работает не бесконечно, а прекращает работу через некоторое время (например, 30 сек). Использовать функцию `clock()` для определения времени работы сервера.

Что будет в случае, если сервер завершит работу, не закрыв канал?

## Выполнение лабораторной работы

---




Изучила материал лабораторной работы. На основе примеров написала аналогичные программы, внося некоторые изменения.

Перед началом выполнения лабораторной работы создала каталог lab14 и файлы: common.h, server.c, client.c, Makefile.

```
[emmednikova@fedora ~]$ mkdir lab14
[emmednikova@fedora ~]$ cd lab14/
[emmednikova@fedora lab14]$ touch common.h server.c client.c Makefile
[emmednikova@fedora lab14]$ ls
client.c  common.h  Makefile  server.c
[emmednikova@fedora lab14]$
```

Вставила текст программы в файл common.h, ничего не меняя.

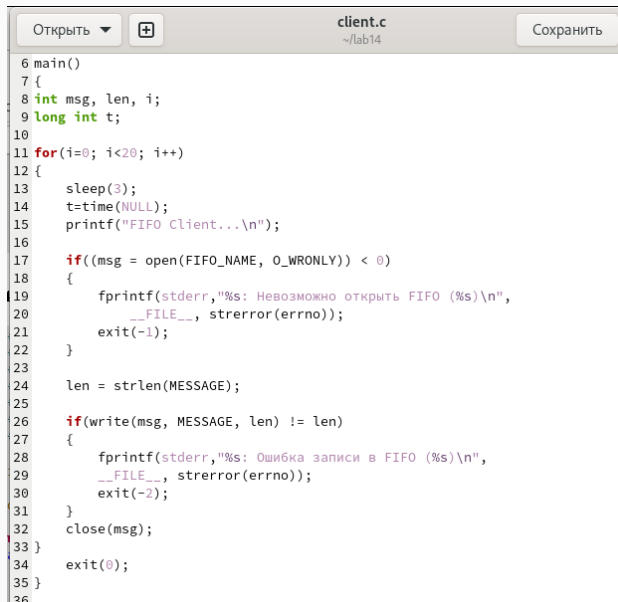
Открыть ▼  \*common.h  
~/lab14 Сохранить

```
1 /*
2 2 * common.h - заголовочный файл со стандартными определениями
3 3 */
4
5 #ifndef __COMMON_H__
6 #define __COMMON_H__
7
8 #include <stdio.h>
9 #include <stdlib.h>
10 #include <string.h>
11 #include <errno.h>
12 #include <sys/types.h>
13 #include <sys/stat.h>
14 #include <fcntl.h>
15
16 #define FIFO_NAME "/tmp/fifo"
17 #define MAX_BUFF 80
18
19 #endif /* __COMMON_H__ */
```

Вставила текст программы в файл client.c, изменила его.

```
Открыть + client.c ~/lab14 Сохранить
1 #include "common.h"
2
3 #define MESSAGE "Hello Server!!!\n"
4
5 int
6 main()
7 {
8     int msg, len, i;
9     long int t;
10
11     for(i=0; i<20; i++)
12     {
13         sleep(3);
14         t=time(NULL);
15         printf("FIFO Client...\n");
16
17         if((msg = open(FIFO_NAME, O_WRONLY)) < 0)
18         {
19             fprintf(stderr,"%s: Невозможно открыть FIFO (%s)\n",
20                     __FILE__, strerror(errno));
21             exit(-1);
22         }
23
24         len = strlen(MESSAGE);
25
26         if(write(msg, MESSAGE, len) != len)
27         {
28             fprintf(stderr,"%s: Ошибка записи в FIFO (%s)\n",
29                     __FILE__, strerror(errno));
30             exit(-2);
31         }
32     }
33 }
```

Вставила текст программы в файл client.c, изменила его.



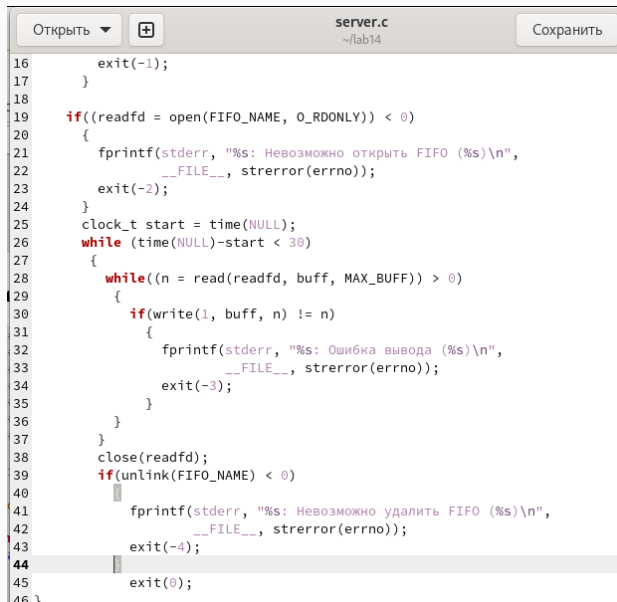
```
6 main()
7 {
8     int msg, len, i;
9     long int t;
10
11     for(i=0; i<20; i++)
12     {
13         sleep(3);
14         t=time(NULL);
15         printf("FIFO Client...\n");
16
17         if((msg = open(FIFO_NAME, O_WRONLY)) < 0)
18         {
19             fprintf(stderr,"%s: Невозможно открыть FIFO (%s)\n",
20                     __FILE__, strerror(errno));
21             exit(-1);
22         }
23
24         len = strlen(MESSAGE);
25
26         if(write(msg, MESSAGE, len) != len)
27         {
28             fprintf(stderr,"%s: Ошибка записи в FIFO (%s)\n",
29                     __FILE__, strerror(errno));
30             exit(-2);
31         }
32         close(msg);
33     }
34     exit(0);
35 }
```

Вставила текст программы в файл server.c, добавив некоторые корректировки.

```
server.c
~/lab14
Сохранить


1 #include "common.h"
2
3 int
4 main()
5 {
6     int readfd; /* дескриптор для чтения из FIFO */
7     int n;
8     char buff[MAX_BUFF]; /* буфер для чтения данных из FIFO */
9
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",
15                __FILE__, strerror(errno));
16        exit(-1);
17    }
18
19    if((readfd = open(FIFO_NAME, O_RDONLY)) < 0)
20    {
21        fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",
22                __FILE__, strerror(errno));
23        exit(-2);
24    }
25    clock_t start = time(NULL);
26    while (time(NULL)-start < 30)
27    {
28        while((n = read(readfd, buff, MAX_BUFF)) > 0)
29        {
30            if(write(1, buff, n) != n)
31            {
```

Вставила текст программы в файл server.c, добавив некоторые корректировки.



```
16     exit(-1);
17 }
18
19 if((readfd = open(FIFO_NAME, O_RDONLY)) < 0)
20 {
21     fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",
22             __FILE__, strerror(errno));
23     exit(-2);
24 }
25 clock_t start = time(NULL);
26 while (time(NULL)-start < 30)
27 {
28     while((n = read(readfd, buff, MAX_BUFF)) > 0)
29     {
30         if(write(1, buff, n) != n)
31         {
32             fprintf(stderr, "%s: Ошибка вывода (%s)\n",
33                     __FILE__, strerror(errno));
34             exit(-3);
35         }
36     }
37 }
38 close(readfd);
39 if(unlink(FIFO_NAME) < 0)
40 {
41     fprintf(stderr, "%s: Невозможно удалить FIFO (%s)\n",
42             __FILE__, strerror(errno));
43     exit(-4);
44 }
45 exit(0);
46 }
```

Вставила текст программы в файл Makefile, ничего не меняя.

Открыть ▾ 

Makefile  
~/lab14

```
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
```



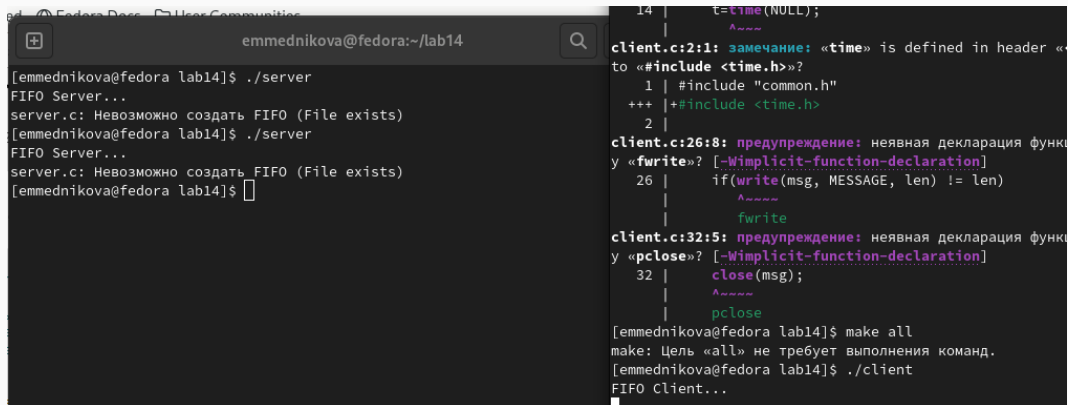
Далее делаем компиляцию файлов с помощью команды `make all`.

```
[emmednikova@fedora lab14]$ make all
gcc server.c -o server
server.c: В функции «main»:
server.c:25:23: предупреждение: неявная декларация функции «time» [-Wimplicit-fu
nction-declaration]
    25 |         clock_t start = time(NULL);
        |                             ^~~~
server.c:2:1: замечание: «time» is defined in header «<time.h>»; did you forget
to «#include <time.h>»?
    1 | #include "common.h"
    ++ |+#include <time.h>
    2 |
server.c:28:21: предупреждение: неявная декларация функции «read»; имелось в вид
е «fread»? [-Wimplicit-function-declaration]
```

Далее делаем компиляцию файлов с помощью команды `make all`.

```
gcc client.c -o client
client.c: В функции «main»:
client.c:13:5: предупреждение: неявная декларация функции «sleep» [-Wimplicit-fun
nction-declaration]
    13 |     sleep(3);
        |     ^~~~~
client.c:14:7: предупреждение: неявная декларация функции «time» [-Wimplicit-fun
ction-declaration]
    14 |     t=time(NULL);
        |     ^~~~
client.c:2:1: замечание: «time» is defined in header «<time.h>»; did you forget
to «#include <time.h>»?
    1 | #include "common.h"
```

При создании и проверке работы файлов каждый терминал (использовалось два терминала) вывел несколько сообщений, после 30 секунд работа сервера была завершена. Если сервер завершит работу, не закрывая канал, то при повторном запуске появится ошибка, так как уже существует один канал.



```
emmednikova@fedora:~/lab14
[emmednikova@fedora lab14]$ ./server
FIFO Server...
server.c: Невозможно создать FIFO (File exists)
[emmednikova@fedora lab14]$ ./server
FIFO Server...
server.c: Невозможно создать FIFO (File exists)
[emmednikova@fedora lab14]$

14 |         t=time(NULL);
    |         ^~~~~~
client.c:2:1: замечание: «time» is defined in header «time.h»
to «#include <time.h>»?
    1 | #include "common.h"
    +++ |+#include <time.h>
    2 |
client.c:26:8: предупреждение: неявная декларация функции «fwrite»? [-Wimplicit-function-declaration]
    26 |         if(write(msg, MESSAGE, len) != len)
        |             ^~~~~~
        |             fwrite
client.c:32:5: предупреждение: неявная декларация функции «pclose»? [-Wimplicit-function-declaration]
    32 |         pclose(msg);
        |             ^~~~~~
        |             pclose
[emmednikova@fedora lab14]$ make all
make: Цель «all» не требует выполнения команд.
[emmednikova@fedora lab14]$ ./client
FIFO Client...
```

## Выводы

---

Приобрела практические навыки работы с именованными каналами.