

# Лабораторная работа №14

Именованные каналы

---

Кузнецова С. В.

30 апреля 2023

Российский университет дружбы народов, Москва, Россия

## Информация

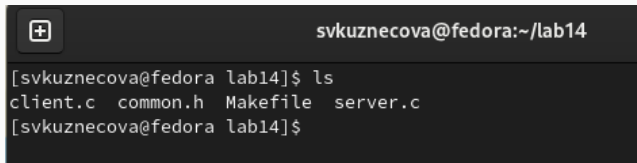
---

- Кузнецова София Вадимовна
- студент прикладной информатики
- Российский университет дружбы народов

## Выполнение лабораторной работы

---

В домашнем каталоге создадим файлы `common.h`, `server.c`, `client.c` и `Makefile`, выполним проверку. Перейдём в `emacs`.

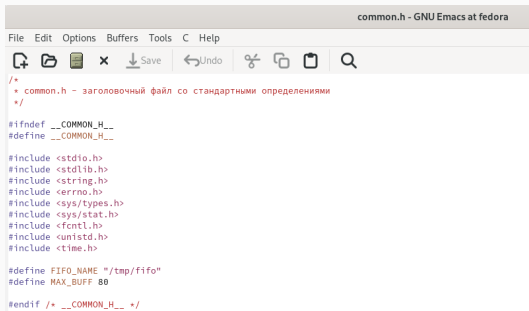
A terminal window with a dark background. The title bar shows a plus icon in a square and the text 'svkuznecova@fedora:~/lab14'. The terminal content shows a user running the 'ls' command in the 'lab14' directory, which lists 'client.c', 'common.h', 'Makefile', and 'server.c'.

```
[svkuznecova@fedora lab14]$ ls
client.c  common.h  Makefile  server.c
[svkuznecova@fedora lab14]$
```

Рис. 1: Создание файлов в домашнем каталоге

## Первый скрипт

В emacs откроем созданный файл common.h и приступим к переносу в него скрипта из файла, а также внесём в него дополнительные изменения.



```
/*
 * common.h - заголовочный файл со стандартными определениями
 */

#ifndef __COMMON_H__
#define __COMMON_H__

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>
#include <time.h>

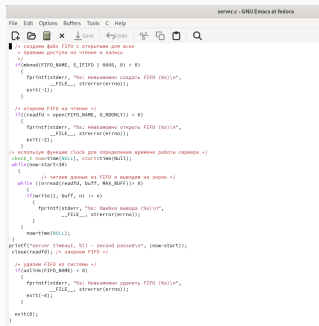
#define FIFO_NAME "/tmp/fifo"
#define MAX_BUFF 80

#endif /* __COMMON_H__ */
```

Рис. 2: Перенос скрипта для common.h

## Второй скрипт

После того как мы перенесли, изменили и сохранили скрипт для первого файла, открываем файл server.c и также переносим в него скрипт с изменениями, но уже для второго файла. Выполняем сохранение.



```
server.c - GNU Emacs at fedora
File Edit Options Buffers Tools C Help
[Icons] [Save] [Undo] [Find] [Search]
/* создаем файл FIFO с открытием для всех
 * правами доступа на чтение и запись
 */
if (mkfifo(FIFO_NAME, S_IFIFO | 0666, 0) < 0)
{
    fprintf(stderr, "%s: невозможно создать FIFO (%s)\n",
            __FILE__, strerror(errno));
    exit(-1);
}

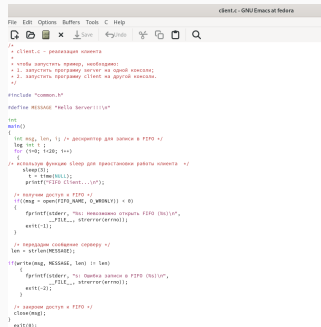
/* открываем FIFO на чтение */
if ((readfd = open(FIFO_NAME, O_RDONLY)) < 0)
{
    fprintf(stderr, "%s: невозможно открыть FIFO (%s)\n",
            __FILE__, strerror(errno));
    exit(-2);
}

/* используем функцию clock для опраивания времени работы сервера */
clock_t nowtime(NULL), starttime(NULL);
while (now-starttime)
{
    /* читаем данные из FIFO и выведем на экран */
    while ((nread=readfd, buff, MAX_BUF)) > 0)
    {
        if (write(1, buff, n) != n)
        {
            fprintf(stderr, "%s: багged вывод (%s)\n",
                    __FILE__, strerror(errno));
        }
    }
    nowtime(NULL);
}
printf("server timeout, %d - second passed\n", (now-starttime));
close(readfd); /* закрываем FIFO */

/* удаляем FIFO из системы */
if (unlink(FIFO_NAME) < 0)
{
    fprintf(stderr, "%s: невозможно удалить FIFO (%s)\n",
            __FILE__, strerror(errno));
    exit(-4);
}
exit(0);
```

Рис. 3: Перенос скрипта для server.c

Теперь нам нужно перенести третий скрипт в файл client.c. После чего также выполняем сохранение.



```
client.c - GNU Emacs at fedora
File Edit Options Buffers Tools C Help
[Icons] Save Undo [Icons] Search

/*
 * client.c - реализация клиента
 *
 * чтобы запустить пример, необходимо:
 * 1. запустить программу server на одной консоли;
 * 2. запустить программу client на другой консоли.
 */

#include "common.h"

#define MESSAGE "Hello Server!!!!"

int
main()
{
    int msg, len, i; /* дескриптор для записи в FIFO */
    long tot = 0;
    for (i=0; i<20; i++)
    {
        /* используем функцию sleep для предоставления работы клиенту */
        sleep(1);
        tot = tot + 1;
        printf("FIFO client...%d\n", tot);
    }

    /* наладим доступ к FIFO */
    if ((msg = open(FIFO_NAME, O_WRONLY)) < 0)
    {
        fprintf(stderr, "Не удалось открыть FIFO (%d)\n",
                __FILE__, strerror(errno));
        exit(-1);
    }

    /* передаем сообщение серверу */
    len = strlen(MESSAGE);
    if (write(msg, MESSAGE, len) != len)
    {
        fprintf(stderr, "Не удалось записать в FIFO (%d)\n",
                __FILE__, strerror(errno));
        exit(-2);
    }

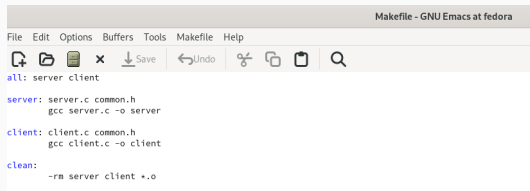
    /* закрываем доступ к FIFO */
    close(msg);
}

exit(0);
```

Рис. 4: Перенос скрипта для client.c



Выполняем перенос скрипта для последнего, четвёртого файла Makefile и закрываем emacs.

The image shows a screenshot of the GNU Emacs editor window titled "Makefile - GNU Emacs at fedora". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Makefile", and "Help". The toolbar contains icons for opening, saving, undo, redo, and search. The main text area displays a Makefile script with the following content:

```
all: server client

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

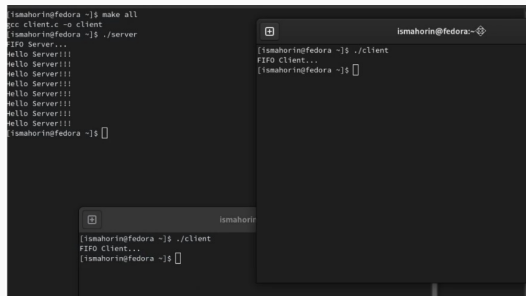
client: client.c common.h
    gcc client.c -o client

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

Рис. 5: Перенос скрипта для Makefile

## Проверка выполнения

В терминале выполняем команду `make all`. После чего в одном терминале запустим команду `./server`, а в других `./client` и проверим корректность выполнения.



```
ismahorin@fedora ~]$ make all
gcc client.c -o client
ismahorin@fedora ~]$ ./server
FIFO Server...
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
ismahorin@fedora ~]$

ismahorin@fedora ~]$ ./client
FIFO Client...
ismahorin@fedora ~]$

ismahorin@fedora ~]$ ./client
FIFO Client...
ismahorin@fedora ~]$
```

Рис. 6: Проверка выполнения

## Выводы

---

В ходе выполнения лабораторной работы были приобретены практические навыки работы с именованными каналами.

Спасибо за внимание!