# Российский университет дружбы народов

### Факультет физико-математических и естественных наук

## Отчёт по лабораторной работе №15

### Именованные каналы

**Дисциплина:** Операционные системы

**Студент:** Оразгелдиева Огулнур

**Группа:** НПИбд-02-20

**Студ. номер:** 1032205431

2021, Москва

**Лабораторная работа №15**

**Именованные каналы**

**Цель:**

Приобрести практические навыки работы с именованными каналами

**Задачи:**

1. Ознакомиться с теоретическим материалом
2. Изучить приведённые в тексте программы server.c и client.c. Взяв данные примеры за образец, написать аналогичные программы, внеся 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, 30сек). Использовать функцию clock() для определения времени работы сервера.

**Теоретические сведения** [[1]](https://esystem.rudn.ru/pluginfile.php/1142389/mod_resource/content/1/013-ipc-fifo.pdf) [[2]](http://cppstudio.com/post/561/)

Одним из видов взаимодействия между процессами в операционных системах является обмен сообщениями. Под сообщением понимается последовательность байтов, передаваемая от одного процесса другому

Для передачи данных между неродственными процессами можно использовать механизм именованных каналов (named pipes). Данные передаются по принципу FIFO (First In First Out) (первым записан — первым прочитан), поэтому они называются также FIFO pipes или просто FIFO. Именованные каналы отличаются от неименованных наличием идентификатора канала, который представлен как специальный файл (соответственно имя именованного канала — это имя файла). Поскольку файл находится на локальной файловой системе, данное IPC используется внутри одной системы.

Файлы именованных каналов создаются функцией mkfifo(3).

include

include

int mkfifo(const char \* pathname, mode\_t mode);

Первый параметр — имя файла, идентифицирующего канал, второй параметр — маска прав доступа к файлу.

Для создания файла FIFO можно использовать более общую функцию mknod(2), предназначенную для создания специальных файлов различных типов (FIFO,сокеты, файлы устройств и обычные файлы для хранения данных).

include

include

include <fcntl.h>

include <unistd.h>

int mknod(const char \*pathname, mode\_t mode, dev\_t dev);

Тогда,вместо >mkfifo(FIFO\_NAME, 0600);

пишем >mknod(FIFO\_NAME, S\_IFIFO | 0600, 0);

Каналы представляют собой простое и удобное средство передачи данных, которое, однако, подходит не во всех ситуациях. Например, с помощью каналов довольно трудно организовать обмен асинхронными сообщениями между процессами. [[1]](https://esystem.rudn.ru/pluginfile.php/1142389/mod_resource/content/1/013-ipc-fifo.pdf)

Дополнительно:

Функция clock() возвращает количество временных тактов, прошедших с начала запуска программы.

С помощью макроса CLOCKS\_PER\_SEC функция получает количество пройденных тактов за 1 секунду. Таким образом, зная сколько выполняется тактов в секунду, зная время запуска программы можно посчитать время работы всей программы или отдельного её фрагмента, что и делает данная функция.

Возвращаемое значение - число тактов прошедшее с момента запуска программы.

Возвращаемое значение функции clock имеет тип данных clock\_t. clock\_t способный представлять временные такты, а также поддерживает арифметические операции. [[2]](http://cppstudio.com/post/561/)

**Ход работы**

1. Ознакомившись с теоретическим материалом, рассмотрим тексты программ файлов server., client.c.
2. Будем пользоваться редактором vi. Создадим с помощью этого редактора файл common.h. (см. рис. 1)

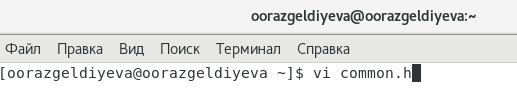


Рисунок 1. Создание файла common.h

Напишем текст программы согласно образцу. В этом файле будут библиотеки, которые будут включены в программу.

Добавим библиотеку <time.h>, которая позволит работать с функцией clock(). (см. рис. 2)

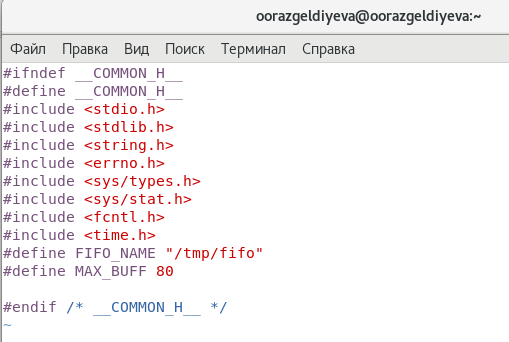


Рисунок 2. common.h

1. Создадим и откроем файл server.c с помощью редактора vi (см. рис. 3), напишем в нем код программы. (см. рис. 4-5)

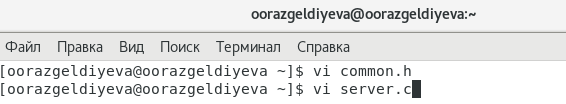


Рисунок 3. Создание файла server.c

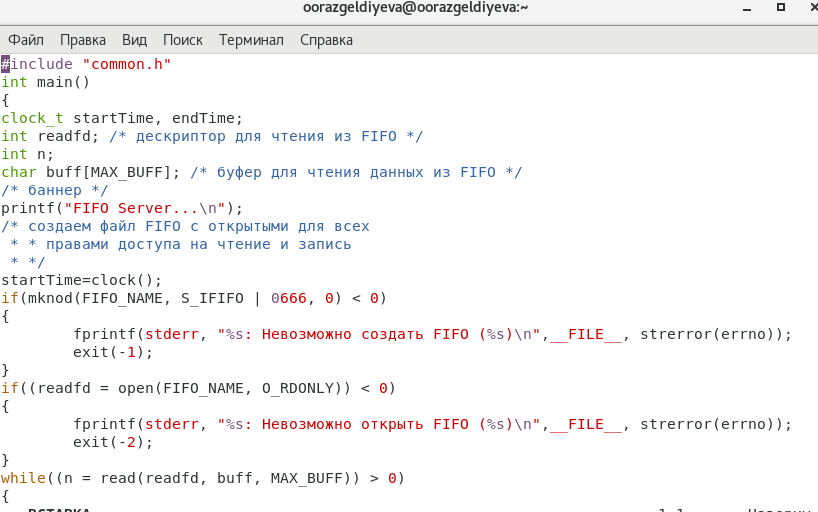


Рисунок 4. server.c

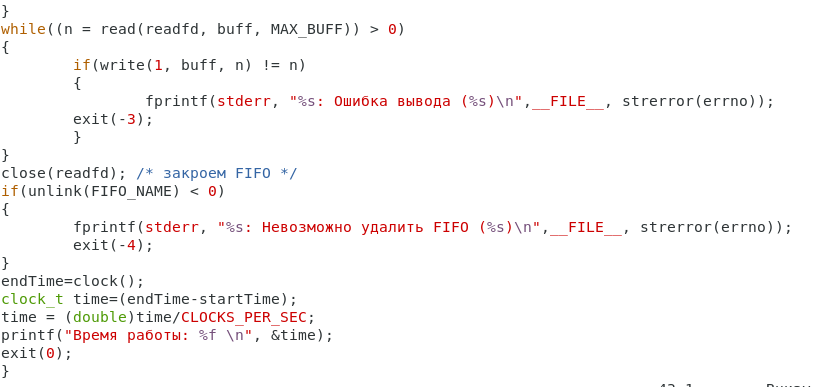


Рисунок 5. server.c

**Пояснения:** с помощью clock\_t зададим переменные для начала и конца работы процесса (startTime, endTime); используя функцию clock() устанавливаем начальное и конечное время (startTime=clock(), endTime=clock()). Записываем и выводим промежуток времени на экран.

1. Создадим и откроем файл client.c внесем в него код программы по образцу. (см. рис. 6)

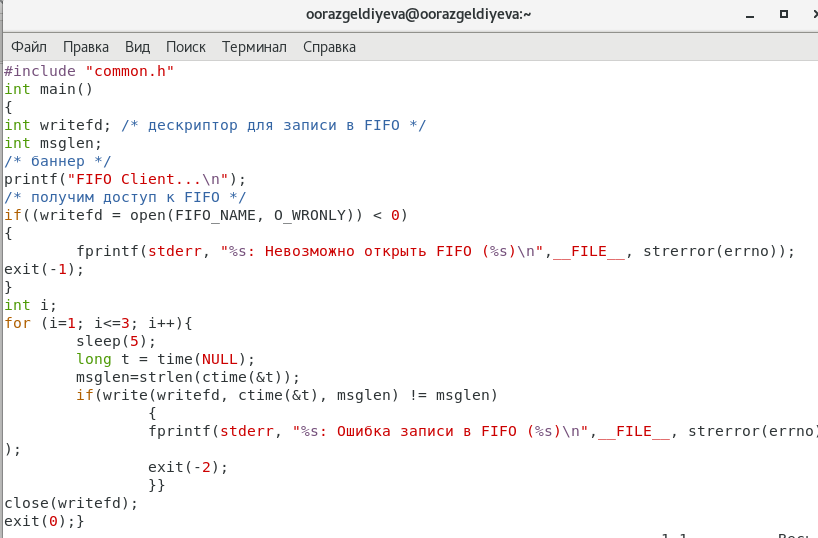


Рисунок 6. client.c

В текст программы заносим изменения.

**Пояснения:** задаем цикл for, который будет каждые 5 секунд выводить сообщение со временем и датой. Выполнятся это действие будет по 3 раза для каждого клиента.

1. Создадим и откроем с помощью радактора vi файл Makefile. Запишем без изменений приведенный в образце текст. (см. рис. 7)

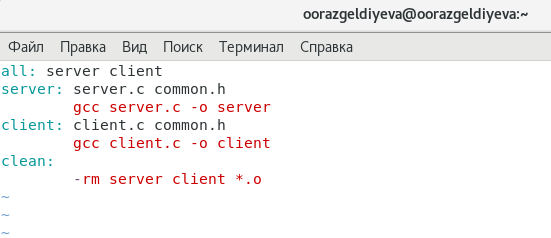


Рисунок 7. Makefile

**Пояснения:** в этом файле заданы цели all, server, client и clean. Цель all будет выполнять цели server и client, которые в свою очередь компилируют файлы server.c и client.c соответственно; цель clean удаляет созданные после компиляции файлы.

1. Используя команду make скомпилируем файлы server.c и client.c. (исползовали цель all). (см. рис. 8)

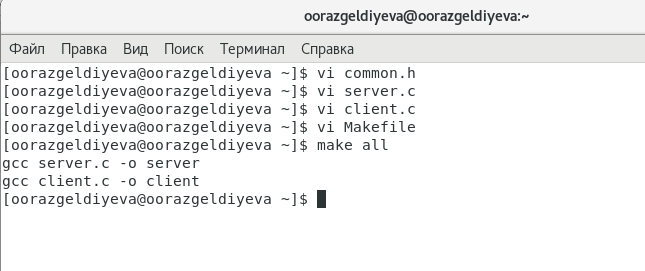


Рисунок 8. Компиляция

1. Запустим в одной консоли программу server, в других двух - client. (см. рис. 9)

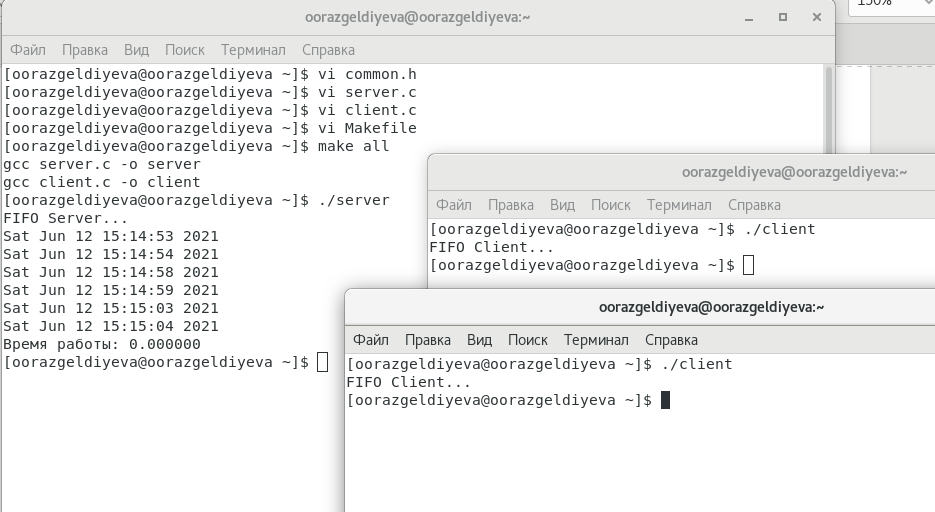


Рисунок 9. Запуск

Как видно из рис. 9, через каждые 5 секунд для каждого клиента выводится дата и время. (например, для первого это в 15:14:53, 15:14:58 и 15:15:03, аналогично со вторым клиентом). В конце выводится время работы сервера.

Если запустить два сервера, то выведется сообщение, что сервер уже существует и невозможно создать его. (см. рис. 10)

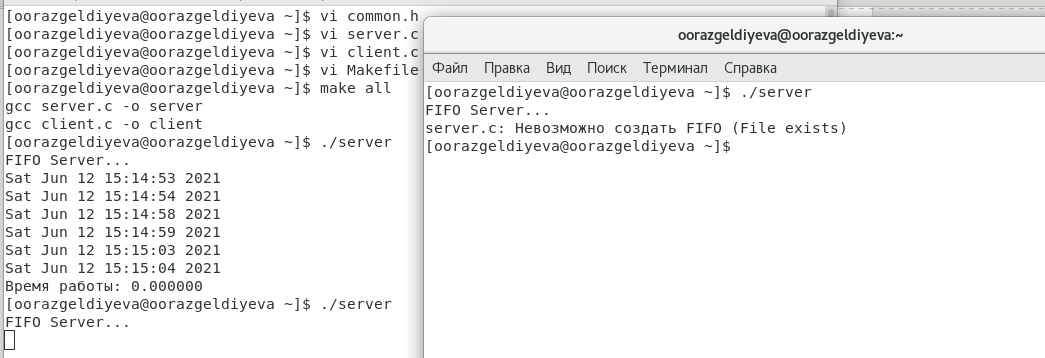


Рисунок 10. Сервер

**Вывод:** на лабораторной работе приобрела практические навыки работы с именованными каналами.

**Библиография**

[[1] - РУДН, дисциплина "Операционные системы", лабораторная работа по теме "Именованные каналы"](https://esystem.rudn.ru/pluginfile.php/1142389/mod_resource/content/1/013-ipc-fifo.pdf)

[[2] - Функция clock(), язык программирования С/С++](http://cppstudio.com/post/561/)