

Лабораторная работа №14

Именованные каналы

Гандич Дарья Владимировна

12 мая 2023 г.

Российский университет дружбы народов, Москва, Россия

Информация

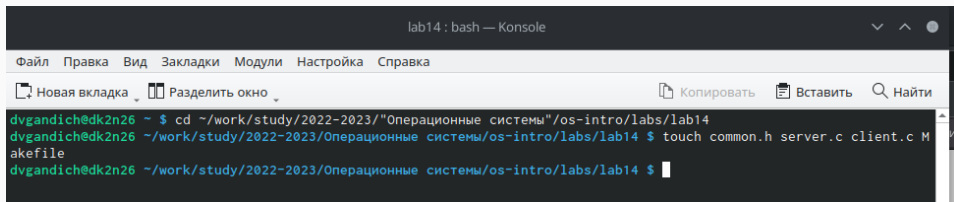
- Гандич Дарья Владимировна
- студентка группы НБИбд-02-22
- Российский университет дружбы народов

Вводная часть

Приобретение практических навыков работы с именованными каналами.

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

2. Создаем файлы common.h, server.c, client.c, Makefile.

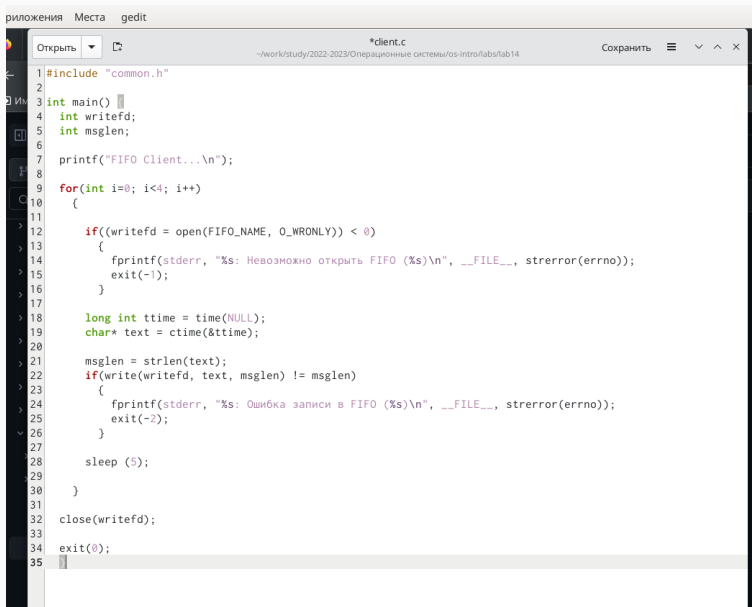


The screenshot shows a terminal window titled "lab14 : bash — Konsole". The window has a menu bar with "Файл", "Правка", "Вид", "Закладки", "Модули", "Настройка", and "Справка". Below the menu bar are icons for "Новая вкладка", "Разделить окно", "Копировать", "Вставить", and "Найти". The terminal content shows the following commands and output:

```
dvgandich@dk2n26 ~ $ cd ~/work/study/2022-2023/"Операционные системы"/os-intro/labs/lab14
dvgandich@dk2n26 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14 $ touch common.h server.c client.c Makefile
dvgandich@dk2n26 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14 $
```

Рис. 1: 1 создание файлов

3. Вбиваем тексты кода в файлы, меняя их по требованию задания



```
приложения Места gedit
*client.c
~/.work/study/2022-2023/Операционные системы/os-intro/labs/lab14
Сохранить

1 #include "common.h"
2
3 int main()
4 {
5     int writefd;
6     int msglen;
7
8     printf("FIFO Client...\n");
9
10    for(int i=0; i<4; i++)
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 }
```


Места gedit

serverc

~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14

Сохранить

```
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);
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 }
```

Рис. 3: server.c

ложения Места gedit

Открыть common.h Сохранить

~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14

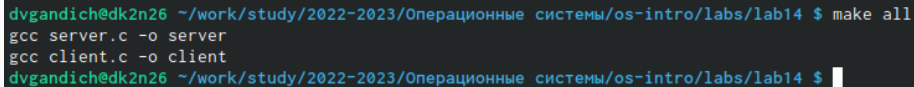
```
1 #ifndef __COMMON_H__
2 #define __COMMON_H__
3
4 #include <stdio.h>
5 #include <stdlib.h>
6 #include <string.h>
7 #include <errno.h>
8 #include <sys/types.h>
9 #include <sys/stat.h>
10 #include <fcntl.h>
11 #include <unistd.h>
12 #include <time.h>
13
14 #define FIFO_NAME "/tmp/fifo"
15 #define MAX_BUFF 80
16
17 #endif
```

Рис. 4: common.h

```
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
```

Рис. 5: Makefile

4. Запускаем все файлы командой make all

A terminal window with a dark background. The prompt is 'dvgandich@dk2n26 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14 \$'. The command 'make all' has been entered. The output shows two compilation commands: 'gcc server.c -o server' and 'gcc client.c -o client'. The prompt is repeated at the bottom, followed by a white cursor block.

```
dvgandich@dk2n26 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14 $ make all
gcc server.c -o server
gcc client.c -o client
dvgandich@dk2n26 ~/work/study/2022-2023/Операционные системы/os-intro/labs/lab14 $
```

Рис. 6: make all

5. Проверяем работу сервера и клиентов

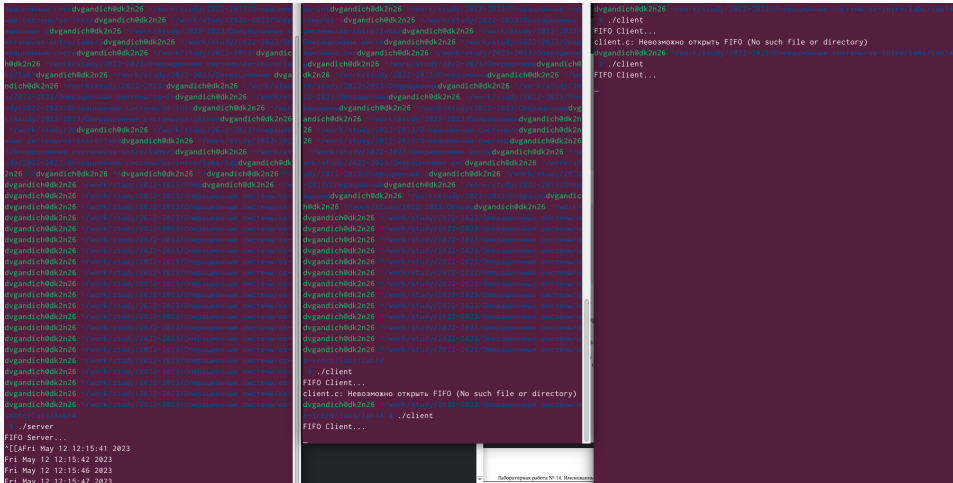


Рис. 7: работа клиентов

- Мы приобрели практические навыки работы с именованными каналами

мы молодцы! :::