

Лабораторная работа No 15

Именованные каналы

Кеан Путхеаро НПИбд-01-20

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
4	Вывод	10
5	Библиография	11

Список иллюстраций

3.1	server	6
3.2	client	7
3.3	client2	7
3.4	common.h	8
3.5	makefile	8
3.6	результат	9

1 Цель работы

Приобретение практических навыков работы с именованными каналами.

2 Задание

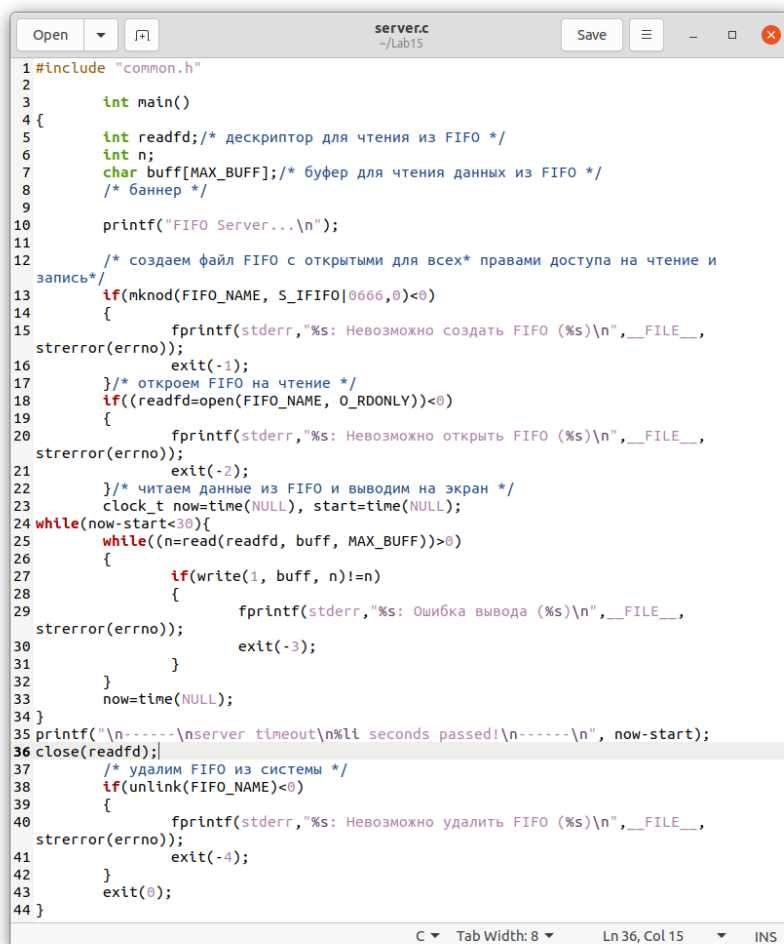
Изучите приведённые в тексте программы `server.c` и `client.c`. Взяв данные примеры за образец, напишите аналогичные программы, внося следующие изменения:

1. Работает не 1 клиент, а несколько (например, два).
2. Клиенты передают текущее время с некоторой периодичностью (например, раз в пять секунд). Используйте функцию `sleep()` для приостановки работы клиента.
3. Сервер работает не бесконечно, а прекращает работу через некоторое время (например, 30 сек). Используйте функцию `clock()` для определения времени работы сервера. Что будет в случае, если сервер завершит работу, не закрыв канал?

3 Выполнение лабораторной работы

Изучил приведённые в тексте программы server.c и client.c.

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



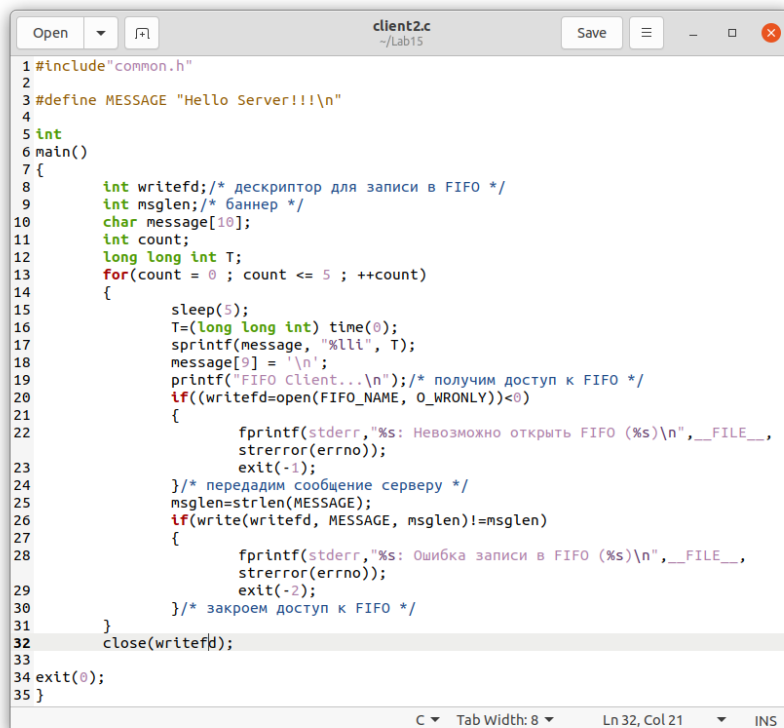
```
1 #include "common.h"
2
3 int main()
4 {
5     int readfd; /* дескриптор для чтения из FIFO */
6     int n;
7     char buff[MAX_BUFF]; /* буфер для чтения данных из FIFO */
8     /* баннер */
9
10    printf("FIFO Server...\n");
11
12    /* создаем файл FIFO с открытыми для всех* правами доступа на чтение и
запись*/
13    if(mknod(FIFO_NAME, S_IFIFO|0666,0)<0)
14    {
15        fprintf(stderr,"%s: Невозможно создать FIFO (%s)\n",__FILE__,
strerror(errno));
16        exit(-1);
17    } /* откроем FIFO на чтение */
18    if((readfd=open(FIFO_NAME, O_RDONLY))<0)
19    {
20        fprintf(stderr,"%s: Невозможно открыть FIFO (%s)\n",__FILE__,
strerror(errno));
21        exit(-2);
22    } /* читаем данные из FIFO и выводим на экран */
23    clock_t now=time(NULL), start=time(NULL);
24    while(now-start<30){
25        while((n=read(readfd, buff, MAX_BUFF))>0)
26        {
27            if(write(1, buff, n)!=n)
28            {
29                fprintf(stderr,"%s: Ошибка вывода (%s)\n",__FILE__,
strerror(errno));
30                exit(-3);
31            }
32        }
33        now=time(NULL);
34    }
35    printf("\n-----\nserver timeout\n%li seconds passed!\n-----\n", now-start);
36    close(readfd);
37    /* удалим FIFO из системы */
38    if(unlink(FIFO_NAME)<0)
39    {
40        fprintf(stderr,"%s: Невозможно удалить FIFO (%s)\n",__FILE__,
strerror(errno));
41        exit(-4);
42    }
43    exit(0);
44 }
```

Рис. 3.1: server



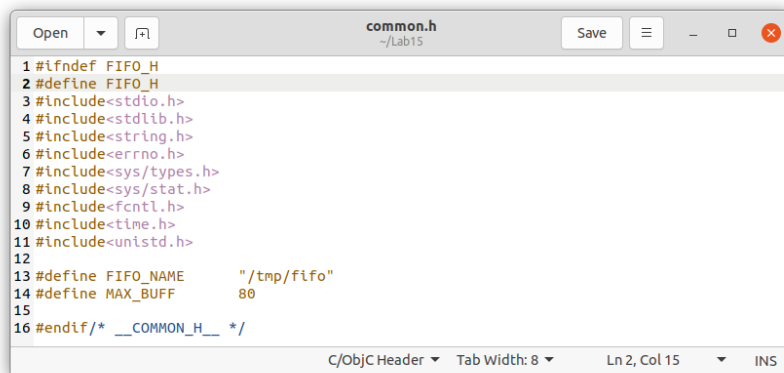
```
1 #include "common.h"
2
3 #define MESSAGE "Hello Server!!!\n"
4
5 int
6 main()
7 {
8     int writefd; /* дескриптор для записи в FIFO */
9     int msglen; /* баннер */
10
11     printf("FIFO Client...\n"); /* получим доступ к FIFO */
12
13     if((writefd=open(FIFO_NAME, O_WRONLY))<0)
14     {
15         fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n", __FILE__,
16             strerror(errno));
17         exit(-1);
18     } /* передадим сообщение серверу */
19     msglen=strlen(MESSAGE);
20     if(write(writefd, MESSAGE, msglen)!=msglen)
21     {
22         fprintf(stderr, "%s: Ошибка записи в FIFO (%s)\n", __FILE__,
23             strerror(errno));
24         exit(-2);
25     } /* закроем доступ к FIFO */
26     close(writefd);
27     exit(0);
28 }
```

Рис. 3.2: client



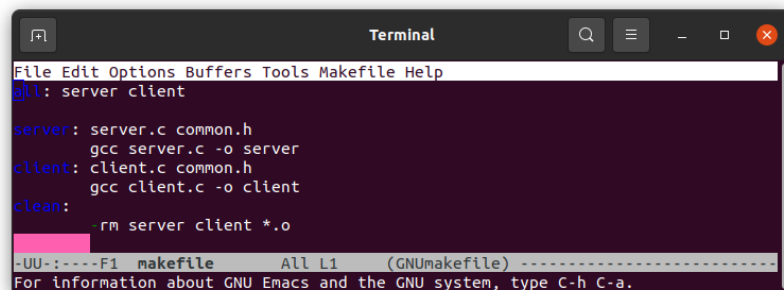
```
1 #include "common.h"
2
3 #define MESSAGE "Hello Server!!!\n"
4
5 int
6 main()
7 {
8     int writefd; /* дескриптор для записи в FIFO */
9     int msglen; /* баннер */
10     char message[10];
11     int count;
12     long long int T;
13     for(count = 0 ; count <= 5 ; ++count)
14     {
15         sleep(5);
16         T=(long long int) time(0);
17         sprintf(message, "%lli", T);
18         message[9] = '\n';
19         printf("FIFO Client...\n"); /* получим доступ к FIFO */
20         if((writefd=open(FIFO_NAME, O_WRONLY))<0)
21         {
22             fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n", __FILE__,
23                 strerror(errno));
24             exit(-1);
25         } /* передадим сообщение серверу */
26         msglen=strlen(MESSAGE);
27         if(write(writefd, MESSAGE, msglen)!=msglen)
28         {
29             fprintf(stderr, "%s: Ошибка записи в FIFO (%s)\n", __FILE__,
30                 strerror(errno));
31             exit(-2);
32         } /* закроем доступ к FIFO */
33     }
34     close(writefd);
35     exit(0);
36 }
```

Рис. 3.3: client2



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

Рис. 3.4: common.h



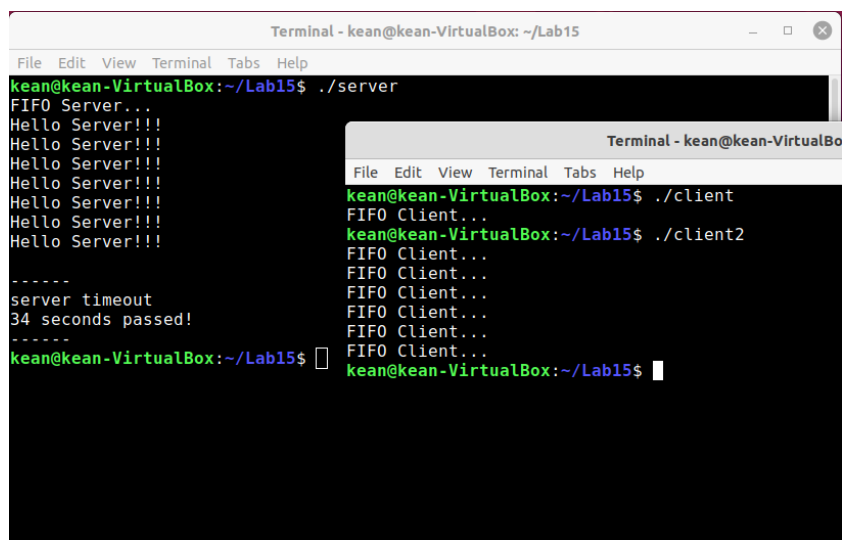
```
File Edit Options Buffers Tools Makefile Help
a)l: server client

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

-UU-:----F1 makefile      All L1      (GNUmakefile) -----
For information about GNU Emacs and the GNU system, type C-h C-a.
```

Рис. 3.5: makefile

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



```
Terminal - kean@kean-VirtualBox: ~/Lab15
File Edit View Terminal Tabs Help
kean@kean-VirtualBox:~/Lab15$ ./server
FIFO Server...
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
-----
server timeout
34 seconds passed!
-----
kean@kean-VirtualBox:~/Lab15$

Terminal - kean@kean-VirtualBox: ~/Lab15
File Edit View Terminal Tabs Help
kean@kean-VirtualBox:~/Lab15$ ./client
FIFO Client...
kean@kean-VirtualBox:~/Lab15$ ./client2
FIFO Client...
FIFO Client...
FIFO Client...
FIFO Client...
FIFO Client...
FIFO Client...
kean@kean-VirtualBox:~/Lab15$
```

Рис. 3.6: результат

3. В случае, если сервер завершит работу, не закрыв канал, файл FIFO не удалится, поэтому его в следующий раз создать будет нельзя и вылезет ошибка, следовательно, работать ничего не будет.

4 Вывод

В результате работы , я приобрел практические навыки работы с именованными каналами

5 Библиография

1. (Лабораторная работа №15) https://esystem.rudn.ru/pluginfile.php/1142389/mod_resource/content/1/ipc-fifo.pdf