

Презентация по лабораторной работе № 14

Королёв И.А.

Российский университет дружбы народов, Москва, Россия

Информация

- Королёв Иван Андреевич
- студент, НКАбд - 05 - 22
- Российский университет дружбы народов

Цель работы

Приобретение практических навыков работы с именованными каналами

Задание

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

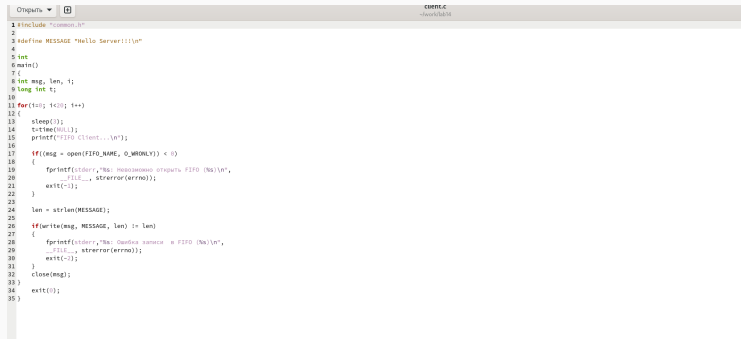
1. Работает не 1 клиент, а несколько (например, два).
2. Клиенты передают текущее время с некоторой периодичностью (например, раз в пятьсекунд). Используйте функцию `sleep()` для приостановки работы клиента.
3. Сервер работает не бесконечно, а прекращает работу через некоторое время (например, 30 сек).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

Теоретическое введение

Одним из видов взаимодействия между процессами в операционных системах является обмен сообщениями. Под сообщением понимается последовательность байтов, передаваемая от одного процесса другому. В операционных системах типа UNIX есть 3 вида межпроцессорных взаимодействий: общедюкские (именованные каналы, сигналы), System V Interface Definition (SVID — разделяемая память, очередь сообщений, семафоры) и BSD (сокеты). 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)`.

Выполнение лабораторной работы

Файл client.c. Создали файл. Сделали, чтобы работал не один клиент, а 2 клиента. Клиенты передают время с периодичностью в 3 секунды.



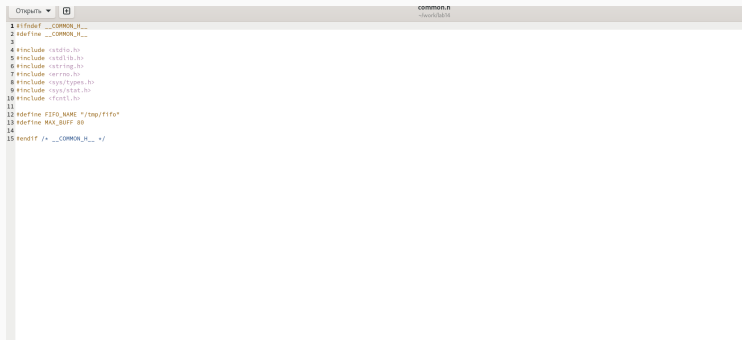
```
1 #include "common.h"
2
3 #define MESSAGE "Hello Server!!!\n"
4
5 int
6 main()
7 {
8     int msg, len, i;
9     long int t;
10
11     for(i=0; i<10; i++)
12     {
13         sleep(3);
14         t=time(NULL);
15         printf("FIFO client...\n");
16
17         if((msg = open(FIFO_NAME, O_WRONLY)) < 0)
18         {
19             fprintf(stderr, "Ns: 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, "Ns: Ошибка записи в FIFO (%s)\n",
29                     _FILE_, strerror(errno));
30             exit(-1);
31         }
32         close(msg);
33     }
34     exit(0);
35 }
```

Рис. 1: client.c

Файл server.c. Создал файл. Сделал, чтобы сервер работал не бесконечно. Прекращает свою работу через 30 секунд.

```
1 #include "common.h"
2 int
3 main()
4 {
5     int readfd;
6     int n;
7     char buff[MAX_BUFF];
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",
13                __FILE__, strerror(errno));
14        exit(-1);
15    }
16
17    if((readfd = open(FIFO_NAME, O_RDONLY)) < 0)
18    {
19        fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",
20                __FILE__, strerror(errno));
21        exit(-2);
22    }
23    clock_t now=time(NULL), start=time(NULL);
24    while(now-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",
31                        __FILE__, strerror(errno));
32            }
33        }
34        now=time(NULL);
35    }
36    printf("server timeout, %li - seconds passed\n", (now-start));
37    close(readfd);
38
39    if(unlink(FIFO_NAME) < 0)
40    {
```

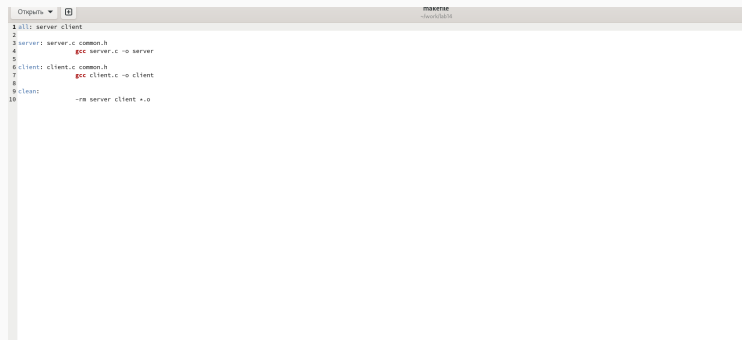
Файл common.h. Создал файл.



The image shows a code editor window with a single file named 'common.h' located at '~\work\lib3d'. The code is a C header file that defines macros for conditional compilation and includes several standard C libraries. The code is as follows:

```
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
12 #define FIFO_NAME "/tmp/fifo"
13 #define MAX_BUFF 80
14
15 #endif /* __COMMON_H__ */
```

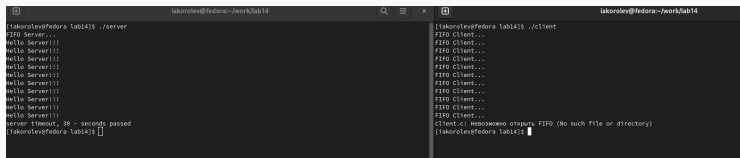
Рис. 3: common.h



The image shows a code editor window with a Makefile. The editor has a menu bar with 'Открыть' (Open) and a toolbar with a file icon. The title bar reads 'makefile' and the path is '~\projects\lab3\3'. The Makefile content is as follows:

```
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. 4: Makefile



```
[takorolev@fedora lab14]$ ./server
FIFO Server...
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
server timeout, 30 - seconds passed
[takorolev@fedora lab14]$
```

```
[takorolev@fedora lab14]$ ./client
FIFO Client...
FIFO Client...
FIFO Client...
FIFO Client...
FIFO Client...
FIFO Client...
FIFO Client...
FIFO Client...
FIFO Client...
FIFO Client...
client.ci: невозможно открыть FIFO (No such file or directory)
[takorolev@fedora lab14]$
```

Рис. 5: Канал

Выводы

Приобрел практические навыки работы с именованными каналами