

Лабораторная работа 15

Акопян Изабелла Арменовна

2021, 11 Июня

RUDN University, Moscow, Russian Federation

Именованные каналы.



- Приобретение практических навыков работы с именованными каналами.

Изучить приведённые в тексте программы `server.c` и `client.c`. Взяв данные примеры за образец, написать 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.

Выполнение лабораторной работы (1/4)

```
1 /*
2 * client.c - реализация клиента
3 *
4 * чтобы запустить процесс, необходимо:
5 * 1. запустить программу server на одной консоли;
6 * 2. запустить программу client на другой консоли.
7 */
8 #include <common.h>
9 #define MESSAGE "Hello Server!!! from 1st"
10 int
11 main()
12 {
13     int writefd; /* дескриптор для записи в FIFO */
14     int maglen;
15     long long int T;
16     for(int i=0; i<10; i++){
17         sleep(1);
18         T+=i;
19         printf("FIFO Client...%d\n", T);
20     }
21     /* получение доступа к FIFO */
22     if((writefd = open(FIFO_NAME, O_WRONLY)) < 0)
23     {
24         fprintf(stderr, "За: невозможно открыть FIFO (%s)\n",
25             _FILE_, strerror(errno));
26         exit(-1);
27     }
28     /* передаем сообщение серверу */
29     maglen = strlen(MESSAGE);
30     if(write(writefd, MESSAGE, maglen) != maglen)
31     {
32         fprintf(stderr, "За: ошибка записи в FIFO (%s)\n",
33             _FILE_, strerror(errno));
34         exit(-1);
35     }
36     /* закрытие доступа к FIFO */
37     close(writefd);
38     exit(0);
39 }
```

```
1 /*
2 * client2.c - реализация клиента
3 *
4 * чтобы запустить процесс, необходимо:
5 * 1. запустить программу server на одной консоли;
6 * 2. запустить программу client на другой консоли.
7 */
8 #include <common.h>
9 #define MESSAGE "Hello Server!!! from 2nd"
10 int
11 main()
12 {
13     int writefd; /* дескриптор для записи в FIFO */
14     int maglen;
15     long long int T;
16     for(int i=0; i<10; i++){
17         sleep(2);
18         T+=i;
19         printf("FIFO Client...%d\n", T);
20     }
21     /* получение доступа к FIFO */
22     if((writefd = open(FIFO_NAME, O_WRONLY)) < 0)
23     {
24         fprintf(stderr, "За: невозможно открыть FIFO (%s)\n",
25             _FILE_, strerror(errno));
26         exit(-1);
27     }
28     /* передаем сообщение серверу */
29     maglen = strlen(MESSAGE);
30     if(write(writefd, MESSAGE, maglen) != maglen)
31     {
32         fprintf(stderr, "За: ошибка записи в FIFO (%s)\n",
33             _FILE_, strerror(errno));
34         exit(-1);
35     }
36     /* закрытие доступа к FIFO */
37     close(writefd);
38     exit(0);
39 }
```

```

Примечание: Места Тестовый редактор
server1.c
Открыть
client.c client2.c common.h Makefile server1.c
1 /*
2 * server.c - реализация сервера
3 *
4 * чтобы запустить процесс, необходимо:
5 * 1. запустить программу server на одной консоли;
6 * 2. запустить программу client на другой консоли.
7 */
8
9 #include "common.h"
10 int
11 main()
12 {
13     int readfd; /* дескриптор для чтения из FIFO */
14     int n;
15     char buff[MAX_BUF]; /* буфер для чтения данных из FIFO */
16     /* Демон */
17     printf("FIFO Server...\n");
18
19     /* создан файл FIFO с открытыми для всех
20      * правами доступа на чтение и запись
21     */
22     if(mknod(FIFO_NAME, S_IFIFO | 0666, 0) < 0)
23     {
24         fprintf(stderr, "Ошибка: невозможно создать FIFO (%s)\n",
25                 _FILE_, strerror(errno));
26         exit(-1);
27     }
28
29     /* открыл FIFO на чтение */
30     if((readfd=open(FIFO_NAME, O_RDONLY)) < 0)
31     {
32         fprintf(stderr, "Ошибка: невозможно открыть FIFO (%s)\n",
33                 _FILE_, strerror(errno));
34         exit(-1);
35     }
36     /* чтение данных из FIFO и выводим на экран */
37     clock_t now = time(NULL), start = time(NULL);
38     while((now-start)<0)
39     {
40         while((n = read(readfd, buff, MAX_BUF)) > 0)
41         {
42             if(write(1, buff, n) != n)
43             {
44                 fprintf(stderr, "Ошибка: вывод (%s)\n",
45                         _FILE_, strerror(errno));
46                 exit(-1);
47             }
48         }
49     }
50 }

```

Курс: Операционные системы | iskorayev@ddk00~1 | iskorayev@ddk00~1 | iskorayev@ddk00~1 | 1 | Настройка файлов | 15.0p | server1.c (r15) - gedit

```

Примечание: Места Тестовый редактор
common.h
Открыть
client.c client2.c common.h Makefile server1.c
1 /*
2 * common.h - заголовочный файл со стандартными определениями
3 */
4
5 #ifndef __COMMON_H__
6 #define __COMMON_H__
7
8 #include <stdio.h>
9 #include <stdlib.h>
10 #include <string.h>
11 #include <sys/types.h>
12 #include <sys/types.h>
13 #include <sys/stat.h>
14 #include <fcntl.h>
15
16 #define FIFO_NAME    "/tmp/fifo"
17 #define MAX_BUF      80
18 #endif /* __COMMON_H__ */

```

(3/4)

[illegible]

The screenshot shows a Windows terminal window with the following content:

```

C:\Users\user> clang -g -c server.c
server.c:37:17: предупреждение: неявная декларация функции -time [-Wimplicit-function-declaration]
    37 |     clock_t new = time(NULL), start = time(NULL);
        |                   ^
server.c:39:15: предупреждение: неявная декларация функции -read; имелось в виду -fread? [-Wimplicit-function-declaration]
    39 |     while((c = read(readfd, buff, MAX_BUFFER)) > 1)
        |                   ^
        |                   fread
server.c:41:11: предупреждение: неявная декларация функции -write; имелось в виду -fwrite? [-Wimplicit-function-declaration]
    41 |         if(write(2, buff, n) != n)
        |           ^
        |           fwrite
server.c:51:51: предупреждение: неявная декларация функции -close; имелось в виду -pclose? [-Wimplicit-function-declaration]
    51 |     close(readfd); // закрывем FIFO
        |           ^
        |           pclose
server.c:56:6: предупреждение: неявная декларация функции -unlink [-Wimplicit-function-declaration]
    56 |     if(unlink(FIFO_NAME) < 0)
        |     ^
gcc client.c -o client
client.c: 8 функции: undeclared
client.c:17:3: предупреждение: неявная декларация функции -sleep [-Wimplicit-function-declaration]
    17 |     sleep(5);
        |     ^
        |     sleep
client.c:18:3: предупреждение: неявная декларация функции -time [-Wimplicit-function-declaration]
    18 |     time(NULL);
        |     ^
        |     time
client.c:31:6: предупреждение: неявная декларация функции -write; имелось в виду -fwrite? [-Wimplicit-function-declaration]
    31 |     if(write(writefd, MSG_SIZE, msglen) != msglen)
        |       ^
        |       fwrite
        |       write
client.c:39:3: предупреждение: неявная декларация функции -close; имелось в виду -pclose? [-Wimplicit-function-declaration]

```

7/9

- Я успешно приобрела практические навыки работы с именованными каналами.

Ссылка1

Ссылка2