

Лабораторная работа №14

Легиньких Галина - группа НФИбд-02-21

20.05.2022

Именованные каналы

Приобретение практических навыков работы с именованными каналами.

1. Изучила теорию по теме и примеры программ `server.c` и `client.c`.
2. Файл `common.h` - заголовочный файл со стандартными определениями (рис. 1)

```
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
```

Figure 1: `common.h`

3. Файл `server.c` - реализация сервера. Чтобы запустить пример, необходимо:

1. запустить программу `server` на одной консоли;
2. запустить программу `client` на другой консоли.(рис. 2)

```
1 #include "common.h"
2 int
3 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     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    if((readfd = open(FIFO_NAME, O_RDONLY)) < 0)
17    {
18        fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",
19            __FILE__, strerror(errno));
20        exit(-2);
21    }
22    clock_t now=time(NULL), start=time(NULL);
23    while(now-start<30)
24    {
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",
30                    __FILE__, strerror(errno));
31            }
32            now=time(NULL);
33        }
34        printf("server timeout, %11 - second passed\n", (now-start));
35        close(readfd); /* закроем FIFO */
36        if(unlink(FIFO_NAME) < 0)
37        {
38            fprintf(stderr, "%s: Невозможно удалить FIFO (%s)\n",
39                __FILE__, strerror(errno));
40            exit(-3);
41        }
42    }
43    exit(0);
44 }
```

Figure 2: `server.c`

4. Файл *client.c* - реализация клиента. Чтобы запустить пример, необходимо:

1. запустить программу *server* на одной консоли;
2. запустить программу *client* на другой консоли.(рис. 3)

```
1#include "common.h"
2#define MESSAGE "Hello Server!!!\n"
3int
4main()
5{
6    int msg,len, i;
7    long int t;
8    for (i=0; i<20; i++)
9    {
10        sleep(3);
11        t=time(NULL);
12        printf("FIFO Clien...\n");
13        if((msg = open(FIFO_NAME, O_WRONLY)) < 0)
14        {
15            fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",
16                __FILE__, strerror(errno));
17            exit(-1);
18        }
19        len = strlen(MESSAGE);
20        if(write(msg, MESSAGE, len) != len)
21        {
22            fprintf(stderr, "%s: Ошибка записи в FIFO (%s)\n",
23                __FILE__, strerror(errno));
24            exit(-2);
25        }
26        close(msg);
27    }
28    exit(0);
29 }
30
```

Figure 3: *client.c*

5. Файл *clien2.c* - реализация клиента. (рис. 4)

```
1 #include "common.h"
2 #define MESSAGE "Hello Server!!\n"
3 int
4 main()
5 {
6     int writefd;
7     int msglen;
8     int count;
9     long long int t;
10    char message[10];
11
12    for (count=0; count<=5; ++count)
13    {
14        sleep(5);
15        t=(long long int) time(0);
16        sprintf(message, "%lli", t);
17        if((writefd = open(FIFO_NAME, O_WRONLY)) < 0)
18        {
19            fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",
20                __FILE__, strerror(errno));
21            exit(-1);
22        }
23        msglen = strlen(MESSAGE);
24        if(write(writefd, MESSAGE, msglen) != msglen)
25        {
26            fprintf(stderr, "%s: Ошибка записи в FIFO (%s)\n",
27                __FILE__, strerror(errno));
28            exit(-2);
29        }
30    }
31    close(writefd);
32    exit(0);
33 }
34
```

Figure 4: clien2.c

6. Makefile (рис. 5)

```
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
11
```

Figure 5: makefile

7. Запускаем make: (рис. 6)

```
ming@kali:~/042024 /? $ make
gcc client.c -o client
client.c:8: функция «main»:
 10 |     sleep(3);
    |     ~~~~~
client.c:11:10: предупреждение: неизвестная декларация функции «time» [-Wimplicit-function-declaration]
 11 |     t=time(NULL);
    |     ~~~~~
client.c:20:16: предупреждение: неизвестная декларация функции «write»; имелось в виду «fwrite»? [-Wimplicit-function-declaration]
 20 |         if(write(msg, MESSAGE, len) != len)
    |         ~~~~~
    |         fwrite
client.c:26:11: предупреждение: неизвестная декларация функции «close»; имелось в виду «pclose»? [-Wimplicit-function-declaration]
 26 |         close(msg);
    |         ~~~~~
    |         pclose
```

Figure 6: make

8. Запускаем программу: (рис. 7)

Приобрела практические навыки работы с именованными каналами.