

Лабораторная работа №14

Именованные каналы

Чемоданова А.А.

Приобретение практических навыков работы с именованными каналами.

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. Что будет в случае, если сервер завершит работу, не закрыв канал?

Выполнение лабораторной работы

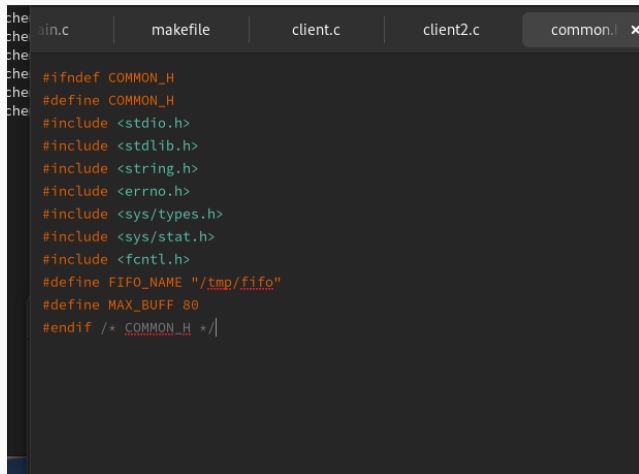
```
1  #include "common.h"
2  #define MESSAGE "Hello Server!!!"
3  int
4  main()
5  {
6      int msg, len, i;
7      long int t2;
8      for(i=0; i<20; i++)
9      {
10         sleep(2);
11         t2=time(NULL);
12         printf("PIDO client...\n");
13
14         if(msg = open(PIPE_NAME, O_WRONLY) < 0)
15         {
16             fprintf(stderr, "%s: невозможно открыть PIPE (%s)\n",
17                     FILE, strerror(errno));
18             exit(1);
19         }
20
21         len = strlen(MESSAGE);
22
23         if(write(msg, MESSAGE, len) != len)
24         {
25             fprintf(stderr, "%s: данные записаны в PIPE (%s)\n",
26                     FILE, strerror(errno));
27             exit(1);
28         }
29         close(msg);
30     }
31     exit(0);
32 }
```

Рис. 1: client.c

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

Рис. 2: client2.c

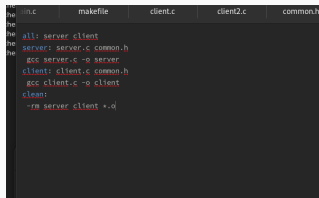
Выполнение лабораторной работы



The image shows a code editor window with several tabs: 'main.c', 'makefile', 'client.c', 'client2.c', and 'common.h'. The 'common.h' tab is active, displaying the following C header file content:

```
#ifndef COMMON_H
#define COMMON_H
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#define FIFO_NAME "/tmp/fifo"
#define MAX_BUFF 80
#endif /* COMMON_H */
```

Рис. 3: common.h



```
all: 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
```

Рис. 4: makefile

```
1 // inc makefile client.c client2.c common.h makefile
2
3 #include "common.h"
4
5 int
6 main()
7 {
8     int readfd;
9     int n;
10    char buff[MAX_BUFF];
11    printf("FIFO Secret...");
12    if(mkfifo(FIFO_NAME, S_IFIFO | 0666, 0) < 0)
13    {
14        fprintf(stderr, "No: %s\n", strerror(errno));
15        exit(-1);
16    }
17    if((readfd = open(FIFO_NAME, O_RDONLY)) < 0)
18    {
19        fprintf(stderr, "No: %s\n", strerror(errno));
20        exit(-2);
21    }
22    clock_t nowTime(NULL), start=TIME(NULL);
23    while(now-start<0)
24    {
25        while((n = read(readfd, buff, MAX_BUFF)) > 0)
26        {
27            if(write(1, buff, n) != n)
28            {
29                fprintf(stderr, "No: %s\n", strerror(errno));
30            }
31        }
32        nowTime(NULL);
33    }
34    printf("server timeout, %ld - seconds passed",
35          (now-start));
36    close(readfd);
37    if(unlink(FIFO_NAME) < 0)
```

Рис. 5: main.c


```
y: ~$ cat server.c  
[aachemodanova@fedora vvv]$ make  
gcc server.c -o server  
server.c: В функции «main»:
```

Рис. 6: Запуск

Мы приобрели практические навыки работы с именованными каналами.