

Лабораторная работа №14

Матюшкин Денис Владимирович (НПИбд-02-21)

01.06.2022

RUDN University, Moscow, Russian Federation

Цель работы

- Приобретение практических навыков работы с именованными каналами.

Ход работы

1.1. Написание программ

- Изучим приведённые в тексте программы `server.c` и `client.c`. Взяв данные примеры за образец, напомним аналогичные программы, внеся следующие изменения:
1. Работает не 1 клиент, а несколько (например, два).
 2. Клиенты передают текущее время с некоторой периодичностью (раз в пять секунд). Используем функцию `sleep()` для приостановки работы клиента.
 3. Сервер работает не бесконечно, а прекращает работу через некоторое время (30 сек). Используем не функцию `clock()`, а взятие `unixTime (time(NULL))`. Дело в том, что `clock()` не засчитывает такты `sleep`. С методом `time()` работает все хорошо. Если сервер завершит свою работу, не закрыв канал, то при повторном запуске сервера будет выводить ошибку: “Невозможно создать FIFO”.

1.2. Иллюстрация работы

- Измененный файл `server.c` (рис. 1 и рис. 2).
- Измененный файл `client.c` (рис. 3 и рис. 4).
- Файл `common.h` (рис. 5).
- Файл `Makefile` (рис. 6).

```

#include "common.h"

int
main()
{
    int readfd; /* дескриптор для чтения из FIFO */
    int n;
    char buff[MAX_BUFF]; /* буфер для чтения данных из FIFO */

    /* баннер */
    printf("FIFO Server...\n");
    /* создаем файл FIFO с открытыми для всех
     * правами доступа на чтение и запись
     */
    if(mknod(FIFO_NAME, S_IFIFO | 0666, 0) < 0)
    {
        fprintf(stderr, "%s: Невозможно создать FIFO (%s)\n",
            __FILE__, strerror(errno));
        exit(-1);
    }

    /* откроем FIFO на чтение */
    if((readfd = open(FIFO_NAME, O_RDONLY)) < 0)
    {
        fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",
            __FILE__, strerror(errno));
        exit(-2);
    }
}

```

Рис. 1: Измененный server.c файл ч.1

```

/*Заведём переменные обозначающие время начала работы сервера и текущее время*/
clock_t now=time(NULL), start=time(NULL);
while(now-start<30){
    /* читаем данные из FIFO и выводим на экран */
    while((n = read(readfd, buff, MAX_BUFF)) > 0)
    {
        if(write(1, buff, n) != n)
        {
            fprintf(stderr, "%s: Ошибка вывода (%s)\n",
                __FILE__, strerror(errno));
            exit(-3);
        }
    }
    now=time(NULL);
}
printf("Времени прошло: %d с.\n", (now-start));
close(readfd); /* закроем FIFO */

/* удалим FIFO из системы */
if(unlink(FIFO_NAME) < 0)
{
    fprintf(stderr, "%s: Невозможно удалить FIFO (%s)\n",
        __FILE__, strerror(errno));
    exit(-4);
}
exit(0);
}

```

Рис. 2: Измененный server.c файл ч.2

```

#include "common.h"

#define MESSAGE "Hello Server!!\n"

int
main()
{
    int writefd; /* дескриптор для записи в FIFO */
    int msglen;
    int i;
    long int ttime;

    for(int i=0; i<5; i++)
    {
        sleep(5);
        ttime=time(NULL);
        /* баннер */
        printf("FIFO Client...\n");
    }
}

```

Рис. 3: Измененный client.c файл ч.1

```

/* получим доступ к FIFO */
if((writefd = open(FIFO_NAME, O_WRONLY)) < 0)
{
    fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",
        __FILE__, strerror(errno));
    exit(-1);
}

/* передадим сообщение серверу */
msglen = strlen(MESSAGE);
if(write(writefd, MESSAGE, msglen) != msglen)
{
    fprintf(stderr, "%s: Ошибка записи в FIFO (%s)\n",
        __FILE__, strerror(errno));
    exit(-2);
}

close(writefd);
}
/* закроем доступ к FIFO */
exit(0);
}

```

Рис. 4: Измененный client.c файл ч.2


```

#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>
#include <time.h>

#define FIFO_NAME "/tmp/fifo"
#define MAX_BUFF 80

#endif /* __COMMON_H__ */

```

Рис. 5: Файл 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. 6: Файл Makefile

2. Создание файлов

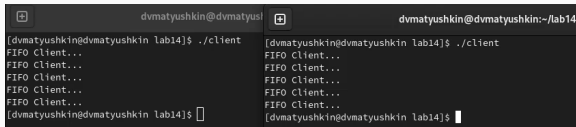
- Пропишем make в консоль, в итоге у нас должны создаваться исполняемые файлы (рис. 7).

```
[dvmatyushkin@dvmatyushkin lab14]$ ls  
client client.c common.h Makefile server server.c  
[dvmatyushkin@dvmatyushkin lab14]$
```

Рис. 7: Использование Makefile

3. Проверка файлов

- Проверим работу файлов (рис. 8 и рис. 9).

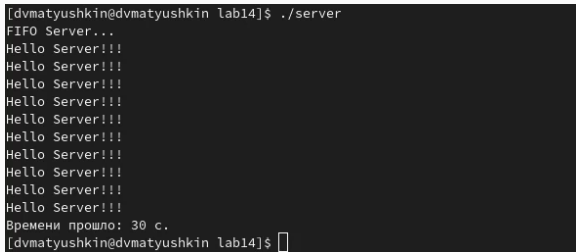


The image shows two side-by-side terminal windows. Both windows have a title bar with a '+' icon and the text 'dvmatyushkin@dvmatyushkin'. The left window's title bar also includes 'lab14'. The left terminal shows the command `./client` being executed, resulting in five lines of output: `FIFO client...`, `FIFO client...`, `FIFO client...`, `FIFO client...`, and `FIFO client...`. The right terminal shows the same command `./client` being executed, resulting in the same five lines of output. Both terminals have a cursor at the end of the last line of output.

```
dvmatyushkin@dvmatyushkin lab14$ ./client
FIFO client...
FIFO client...
FIFO client...
FIFO client...
FIFO client...
[dvmatyushkin@dvmatyushkin lab14]$
```

```
dvmatyushkin@dvmatyushkin:~/lab14$ ./client
FIFO client...
FIFO client...
FIFO client...
FIFO client...
FIFO client...
[dvmatyushkin@dvmatyushkin lab14]$
```

Рис. 8: Проверка client.c файла



The image shows a single terminal window with a title bar containing a '+' icon and the text 'dvmatyushkin@dvmatyushkin lab14'. The terminal shows the command `./server` being executed. The output consists of eight lines of `Hello Server!!!`, followed by a line indicating the elapsed time: `Времени прошло: 30 с.`. The prompt `[dvmatyushkin@dvmatyushkin lab14]$` is visible at the bottom of the terminal.

```
[dvmatyushkin@dvmatyushkin lab14]$ ./server
FIFO Server...
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Hello Server!!!
Времени прошло: 30 с.
[dvmatyushkin@dvmatyushkin lab14]$
```

Рис. 9: Проверка server.c файла

Вывод

- В ходе этой лабораторной работы мы приобрели практические навыки работы с именованными каналами.

Спасибо за внимание!