

Лабораторная работа № 14

Именованные каналы

Шулуужук Айраана Вячеславовна НПИбд-02-22

8 апреля 2023

Российский университет дружбы народов, Москва, Россия

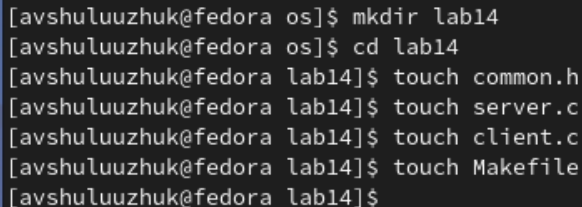
- Шулуужук Айраана Вячеславовна
- НПИбд-02-22
- 1132221890
- Российский университет дружбы народов

Цели и задачи

Приобретение практических навыков работы с именованными каналами.

Выполнение лабораторной работы

Создадим нужный каталог и необходимые файлы

A terminal window with a dark background and light gray text. The prompt is [avshuluuzhuk@fedora os]. The user enters 'mkdir lab14'. The prompt changes to [avshuluuzhuk@fedora lab14]. The user enters 'touch common.h', 'touch server.c', 'touch client.c', and 'touch Makefile' in sequence. The final prompt is [avshuluuzhuk@fedora lab14]\$.

```
[avshuluuzhuk@fedora os]$ mkdir lab14
[avshuluuzhuk@fedora os]$ cd lab14
[avshuluuzhuk@fedora lab14]$ touch common.h
[avshuluuzhuk@fedora lab14]$ touch server.c
[avshuluuzhuk@fedora lab14]$ touch client.c
[avshuluuzhuk@fedora lab14]$ touch Makefile
[avshuluuzhuk@fedora lab14]$
```

Рис. 1: создание файлов и каталога

Выполнение лабораторной работы

Внесем тексты программ в файлы

```
/*  
 * common.h - заголовочный файл со стандартными определениями  
 */  
  
#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. 2: файл commom.h

Выполнение лабораторной работы

В файле server.c введем дополнительную функцию clock для определения времени работы сервера

```
    }  
/* функция clock для определения времени работы сервера */  
  
    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));  
            }  
        }  
        now=time(NULL)  
    }  
    printf("server timeout, %li - second passed\n", (now-start));  
    close(readfd); /* закроем FIFO */
```

Рис. 3: файл server.c

Выполнение лабораторной работы

В файле client.c введем некоторые изменения, клиенты передают текущее время с некоторой периодичностью (например, раз в пять секунд). Используем функцию sleep

```
*/  
#include "common.h"  
  
#define MESSAGE "Hello Server!!!\n"  
  
int  
main()  
{  
    int msg, len, i; /* дескриптор для записи в FIFO */  
    long int i;  
    for(i=0; i<20; i++)  
    {  
        /*функция sleep для приостановки работы клиента*/  
        sleep(5);  
        t = time(NULL)  
        printf("FIFO client...\n");  
        if((msg = open(FIFO_NAME, O_WRONLY))<0)  
        {  
            fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",  
                __FILE__, strerror(errno));  
            exit(-1);  
        }  
        /* передадим сообщение серверу */  
        len = strlen(MESSAGE);  
        if(write(msg, MESSAGE, len) != len)  
        {  
            fprintf(stderr, "%s: Ошибка записи в FIFO (%s)\n",  
                __FILE__, strerror(errno));  
            exit(-2);  
        }  
        /* закроем доступ к FIFO */  
        close(msg);  
    }  
    exit(0);  
}
```

Рис. 4: файл client.c

Создаем файл Makefile

```
ll: 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. 5: Makefile

Скомпилируем программу, используя Makefile

```
[avshuluuzhuk@fedora lab14]$ make
gcc client.c -o client
client.c: В функции «main»:
client.c:21:17: предупреждение: неявная декларация функции «sleep» [-Wimplicit-function-declaration]
    21 |             sleep(3);
        |             ^~~~~~
client.c:22:21: предупреждение: неявная декларация функции «time» [-Wimplicit-function-declaration]
    22 |             t = time(NULL);
        |             ^~~~~~
client.c:10:1: замечание: «time» is defined in header «<time.h>»; did you forget to «#include <time.h>»?
    9 | #include "common.h"
+++ |+#include <time.h>
   10 |
client.c:32:20: предупреждение: неявная декларация функции «write»; имелось в виду «fwrite»? [-Wimplicit-function-declaration]
    32 |             if(write(msg, MESSAGE, len) != len)
        |             ^~~~~~
        |             fwrite
client.c:39:17: предупреждение: неявная декларация функции «close»; имелось в виду «pclose»? [-Wimplicit-function-declaration]
    39 |             close(msg);
        |             ^~~~~~
        |             pclose
```

Рис. 6: компилирование

Запустим командные файлы server и client

<pre>[avshuluuzhuk@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 - second passed [avshuluuzhuk@fedora lab14]\$</pre>	<pre>[avshuluuzhuk@fedora lab14]\$./client FIFO client... FIFO client... FIFO client... FIFO client... FIFO client... FIFO client... FIFO client... FIFO client... FIFO client... FIFO client... FIFO client... client.c: Невозможно открыть FIFO (No such file or directory) [avshuluuzhuk@fedora lab14]\$</pre>
---	--

Рис. 7: запуск командных файлов

Выводы

В ходе выполнения работы мы приобрели практические навыки работы с именованными каналами.