

Презентация по лабораторной работе №14

Именованные каналы

Ибатулина Д.Э.

06 мая 2023

Российский университет дружбы народов, Москва, Россия

Информация

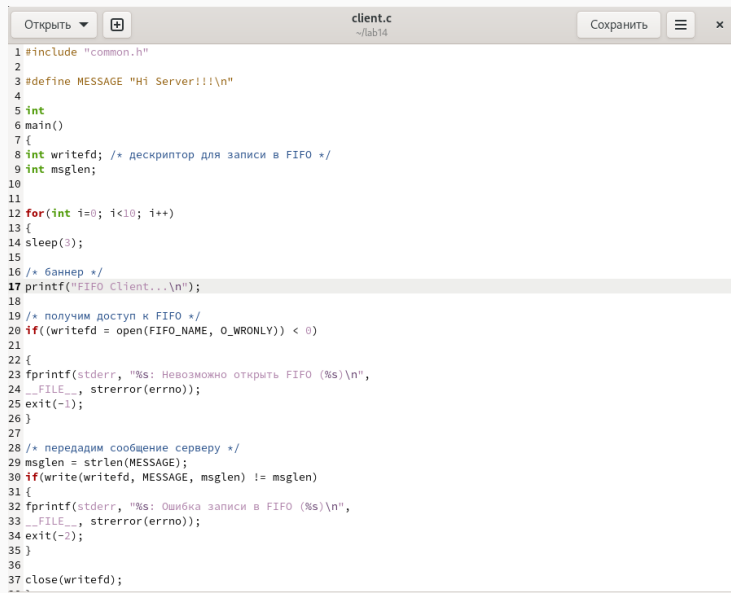
- Ибатулина Дарья Эдуардовна
- студентка группы НКАбд-01-22
- Российский университет дружбы народов
- 1132226434@pfur.ru
- <https://deibatulina.github.io/ru/>

Вводная часть

Умение работать с именованными каналами - очень важный навык для любого IT-специалиста.

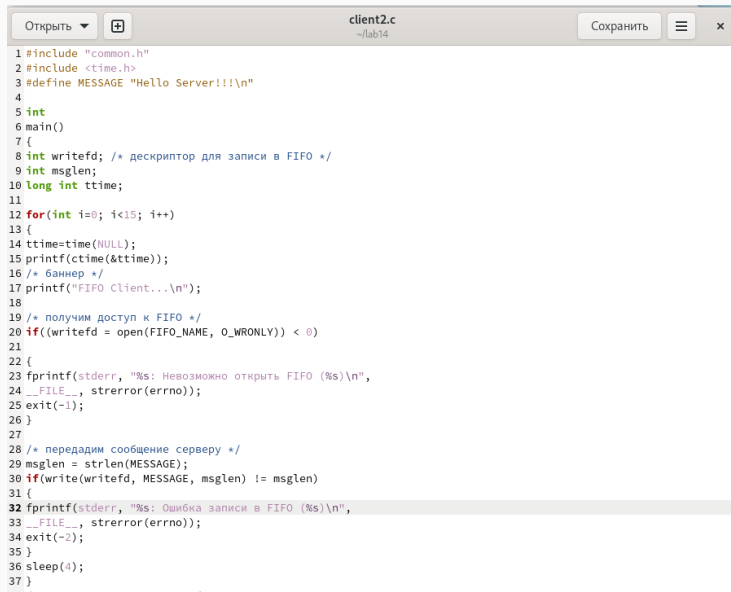
Приобретение практических навыков работы с именованными каналами.

Основная часть



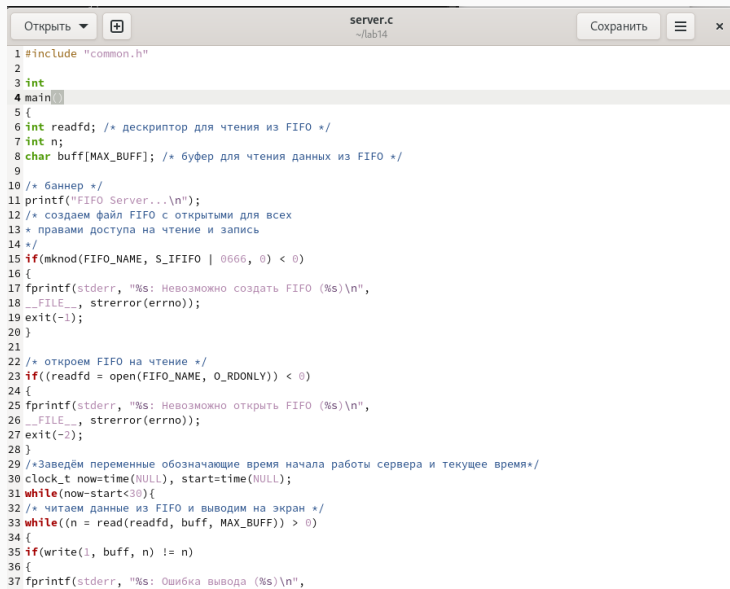
```
1 #include "common.h"
2
3 #define MESSAGE "Hi Server!!!\n"
4
5 int
6 main()
7 {
8     int writefd; /* дескриптор для записи в FIFO */
9     int msglen;
10
11
12     for(int i=0; i<10; i++)
13     {
14         sleep(3);
15
16         /* баннер */
17         printf("FIFO Client...\n");
18
19         /* получим доступ к FIFO */
20         if((writefd = open(FIFO_NAME, O_WRONLY)) < 0)
21         {
22             fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",
23                 __FILE__, strerror(errno));
24             exit(-1);
25         }
26
27         /* передадим сообщение серверу */
28         msglen = strlen(MESSAGE);
29         if(write(writefd, MESSAGE, msglen) != msglen)
30         {
31             fprintf(stderr, "%s: Ошибка записи в FIFO (%s)\n",
32                 __FILE__, strerror(errno));
33             exit(-2);
34         }
35     }
36
37     close(writefd);
38 }
```


Файл client2.c



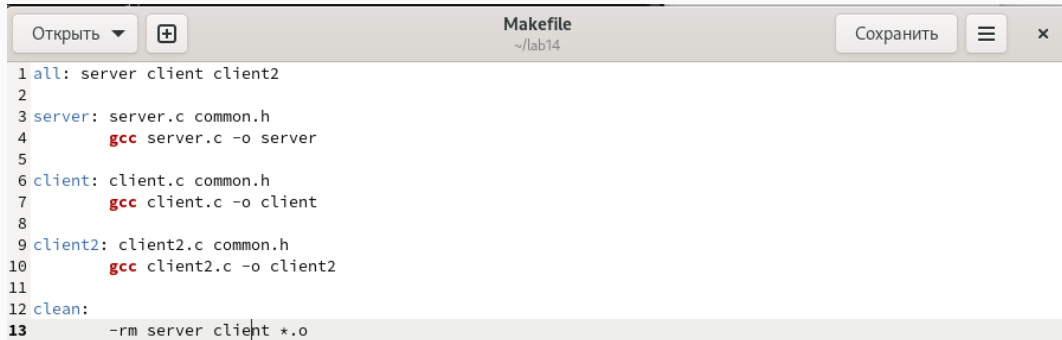
```
1 #include "common.h"
2 #include <time.h>
3 #define MESSAGE "Hello Server!!!\n"
4
5 int
6 main()
7 {
8     int writefd; /* дескриптор для записи в FIFO */
9     int msglen;
10    long int ttime;
11
12    for(int i=0; i<15; i++)
13    {
14        ttime=time(NULL);
15        printf(ctime(&ttime));
16        /* баннер */
17        printf("FIFO Client...\n");
18
19        /* получим доступ к FIFO */
20        if((writefd = open(FIFO_NAME, O_WRONLY)) < 0)
21        {
22            fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",
23                __FILE__, strerror(errno));
24            exit(-1);
25        }
26
27        /* передадим сообщение серверу */
28        msglen = strlen(MESSAGE);
29        if(write(writefd, MESSAGE, msglen) != msglen)
30        {
31            fprintf(stderr, "%s: Ошибка записи в FIFO (%s)\n",
32                __FILE__, strerror(errno));
33            exit(-2);
34        }
35        sleep(4);
36    }
37 }
```

Файл server.c

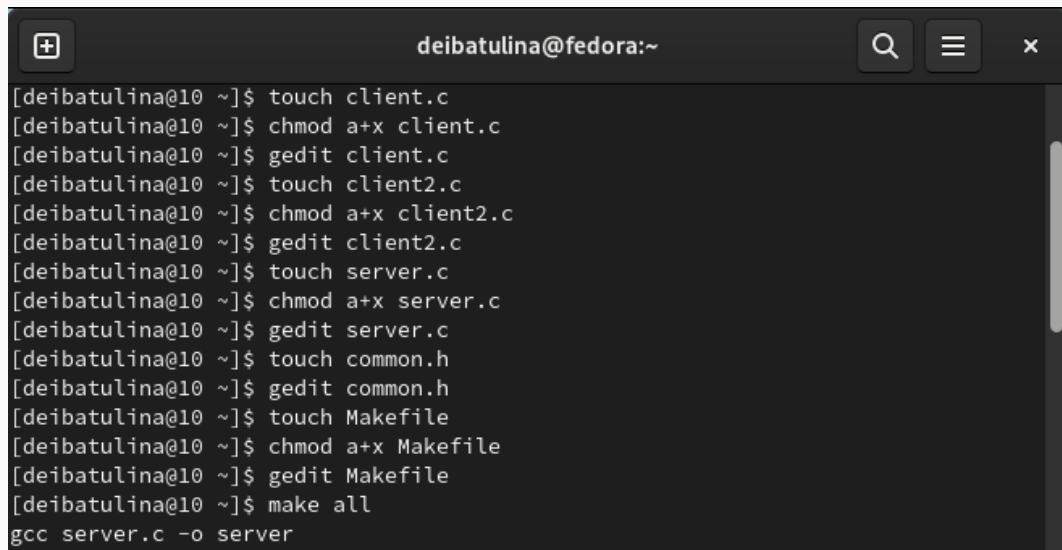


```
1 #include "common.h"
2
3 int
4 main()
5 {
6     int readfd; /* дескриптор для чтения из FIFO */
7     int n;
8     char buff[MAX_BUFF]; /* буфер для чтения данных из FIFO */
9
10    /* баннер */
11    printf("FIFO Server...\n");
12    /* создаем файл FIFO с открытыми для всех
13     * правами доступа на чтение и запись
14     */
15    if(mknod(FIFO_NAME, S_IFIFO | 0666, 0) < 0)
16    {
17        fprintf(stderr, "%s: Невозможно создать FIFO (%s)\n",
18            __FILE__, strerror(errno));
19        exit(-1);
20    }
21
22    /* откроем FIFO на чтение */
23    if((readfd = open(FIFO_NAME, O_RDONLY)) < 0)
24    {
25        fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n",
26            __FILE__, strerror(errno));
27        exit(-2);
28    }
29    /*Заведём переменные обозначающие время начала работы сервера и текущее время*/
30    clock_t now=time(NULL), start=time(NULL);
31    while(now-start<30){
32        /* читаем данные из FIFO и выводим на экран */
33        while((n = read(readfd, buff, MAX_BUFF)) > 0)
34        {
35            if(write(1, buff, n) != n)
36            {
37                fprintf(stderr, "%s: Ошибка вывода (%s)\n",
```

Makefile



```
1 all: server client client2
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 client2: client2.c common.h
10    gcc client2.c -o client2
11
12 clean:
13     rm server client *.o
```



A terminal window titled "deibatulina@fedora:~" with search, menu, and close buttons in the title bar. The terminal displays a series of commands and their outputs for creating and compiling a program.

```
[deibatulina@10 ~]$ touch client.c
[deibatulina@10 ~]$ chmod a+x client.c
[deibatulina@10 ~]$ gedit client.c
[deibatulina@10 ~]$ touch client2.c
[deibatulina@10 ~]$ chmod a+x client2.c
[deibatulina@10 ~]$ gedit client2.c
[deibatulina@10 ~]$ touch server.c
[deibatulina@10 ~]$ chmod a+x server.c
[deibatulina@10 ~]$ gedit server.c
[deibatulina@10 ~]$ touch common.h
[deibatulina@10 ~]$ gedit common.h
[deibatulina@10 ~]$ touch Makefile
[deibatulina@10 ~]$ chmod a+x Makefile
[deibatulina@10 ~]$ gedit Makefile
[deibatulina@10 ~]$ make all
gcc server.c -o server
```

Итоговая часть

В результате выполнения данной лабораторной работы я приобрела практические навыки работы с именованными каналами.

Навыки, приобретённые мною в ходе выполнения данной лабораторной работы, пригодятся мне не только в лабораторных работах, но и в будущей профессии.