

Отчет по лабораторной работе №14

Дисциплина: оперативные системы

Старикова Евгения Дмитриевна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	9

Список иллюстраций

Список таблиц

1 Цель работы

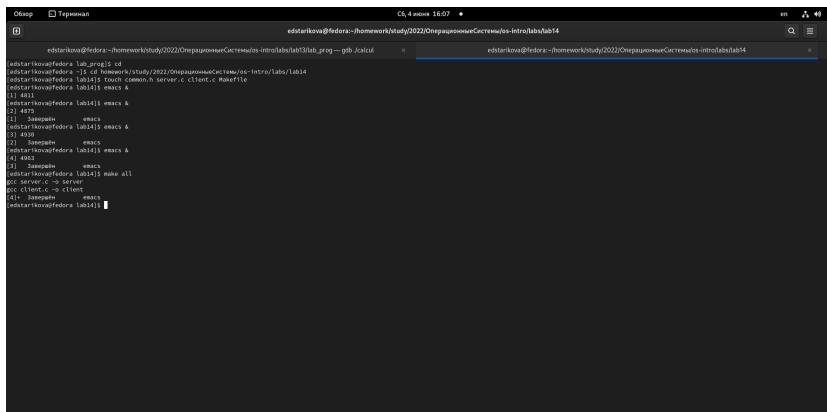
Приобретение практических навыков работы с именованными каналами.

2 Задание

1. Работает не 1 клиент, а несколько (например, два).
2. Клиенты передают текущее время с некоторой периодичностью (например, раз в пять секунд). Используйте функцию `sleep()` для приостановки работы клиента.
3. Сервер работает не бесконечно, а прекращает работу через некоторое время (например, 30 сек). Используйте функцию `clock()` для определения времени работы сервера.

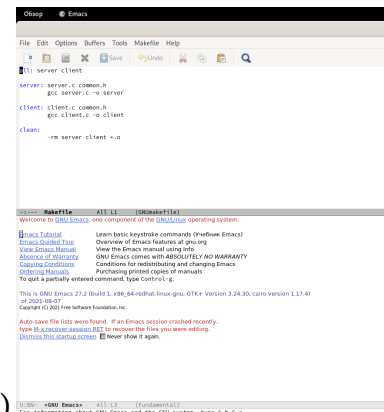
3 Выполнение лабораторной работы

1. (рис. ??)



```
edstarkova@fedora:~/homework/study/2022/ОперационныеСистемы/bs-intro/labs/lab14$ cd
edstarkova@fedora:~/homework/study/2022/ОперационныеСистемы/bs-intro/labs/lab14$ touch common.h server.c client.c Makefile
edstarkova@fedora:~/homework/study/2022/ОперационныеСистемы/bs-intro/labs/lab14$ make
cc -o server server.c
cc -o client client.c
edstarkova@fedora:~/homework/study/2022/ОперационныеСистемы/bs-intro/labs/lab14$ ./server
server: server.c:~$
edstarkova@fedora:~/homework/study/2022/ОперационныеСистемы/bs-intro/labs/lab14$ ./client
client: client.c:~$
edstarkova@fedora:~/homework/study/2022/ОперационныеСистемы/bs-intro/labs/lab14$
```

2. (рис. ??)

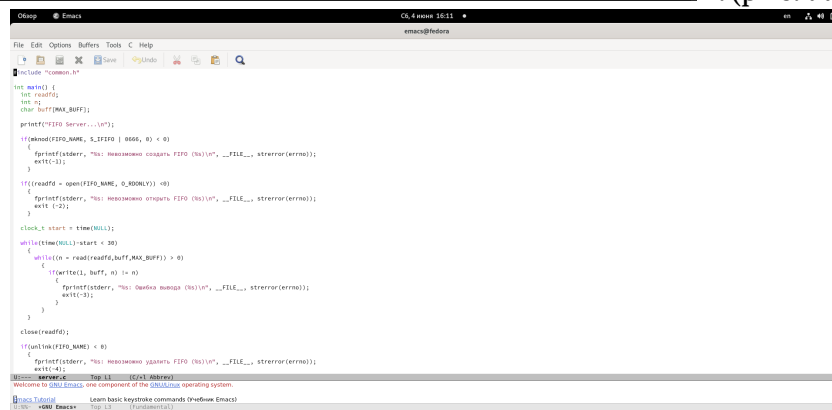


```
server: server.c common.h
gcc server.c -o server

client: client.c common.h
gcc client.c -o client

clean:
rm server client *.o
```

3. (рис. ??)



```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <string.h>
#include <errno.h>
#include <time.h>

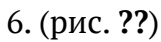
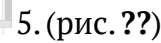
int main() {
    int readfd;
    int writefd;
    char buff[1024];

    printf("FIFO Server...\n");
    if (mkfifo("server", 0666) < 0) {
        perror("mkfifo");
        exit(1);
    }

    if (readfd = open("server", O_RDONLY) < 0) {
        perror("open");
        exit(2);
    }

    clock_t start = time(NULL);
    while (1) {
        int n = read(readfd, buff, sizeof(buff));
        if (n < 0) {
            perror("read");
            exit(3);
        }
        printf("Server: %s\n", buff);
        if (writefd = open("client", O_WRONLY) < 0) {
            perror("open");
            exit(4);
        }
        if (write(writefd, buff, n) < 0) {
            perror("write");
            exit(5);
        }
        close(readfd);
        if (close(writefd) < 0) {
            perror("close");
            exit(6);
        }
    }
}
```

4. (рис. ??)



4 Выводы

Преобрела практические навыки с именными каналами