

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ЛАБОРАТОРНАЯ РАБОТА №14

дисциплина: *Операционные системы*

Студент: Султанова Лейла

Группа: НБИбд-03-22

Москва

2023 г.

- **Цель работы**

Приобретение практических навыков работы с
именованными каналами

1. Создаем необходимые файлы с помощью команды `touch` и открываем редактор `emacs` для их редактирования.

```
[lsultanova@lsultanova ~]$ touch common.h server.c clientc Makefile  
[lsultanova@lsultanova ~]$
```

2. Мы меняем код программы.

В файл `common.h` добавила стандартные заголовочные файлы `unistd.h` и `time.h`, необходимые для работы кодов других файлов. `Common.h` предназначен для заголовочных файлов, чтобы в остальных программах их не прописывать каждый раз.

В файл `server.c` добавила цикл `while` для контроля за временем работы сервера. Разница между текущим временем `time(NULL)` и временем начала работы `clock_t start=time(NULL)` (инициализация до цикла) не должна превышать 30 секунд.

В файл `client.c` добавил цикл, который отвечает за количество сообщений о текущем времени (4 сообщения), которое получается в результате выполнения команд, и команду `sleep(5)` для приостановки работы клиента на 5 секунд.

`Makefile` оставляем без изменений.

```
#include "common.h"

int main() {
    int readfd;
    int n;
    char buff[MAX_BUFF];

    printf("FIFO Server...\n");

    if(mknod(FIFO_NAME, S_IFIFO | 0666, 0) < 0)
    {
        fprintf(stderr, "%s: Невозможно создать FIFO (%s)\n", __FILE__, strerror(errno));
        exit(-1);
    }

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

    clock_t start = time(NULL);

    while(time(NULL)-start<30)
    {
        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);
            }
        }
    }

    close(readfd);
}
```

```
#include "common.h"

int main() {
    int writefd;
    int msglen;

    printf("FIFO Client...\n");

    for(int i=0; i<4; i++)
    {
        if((writefd = open(FIFO_NAME, O_WRONLY)) < 0)
        {
            fprintf(stderr, "%s: Невозможно открыть FIFO (%s)\n", __FILE__, strerror(errno));
            exit(-1);
        }

        long int ttime = time(NULL);
        char* text = ctime(&ttime);

        msglen = strlen(MESSAGE);
        if(write(writefd, MESSAGE, msglen) != msglen)
        {
            fprintf(stderr, "%s: Ошибка записи в FIFO (%s)\n", __FILE__, strerror(errno));
            exit(-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 <unistd.h>
#include <time.h>

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

#endif
```

```
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
```

Вывод

Я приобрела практические навыки работы с именованными каналами.