Лабораторная работа №14

Именованные каналы

Панченко Денис Дмитриевич

Содержание

# 1 Цель работы

Приобретение практических навыков работы с именованными каналами.

# 2 Выполнение лабораторной работы

Создадим файлы: common.h, server.c, client.c, Makefile (рис. [1](#fig:001)).

Figure 1: Создание файлов

Figure 1: Создание файлов

Файл common.h (рис. [2](#fig:002)).

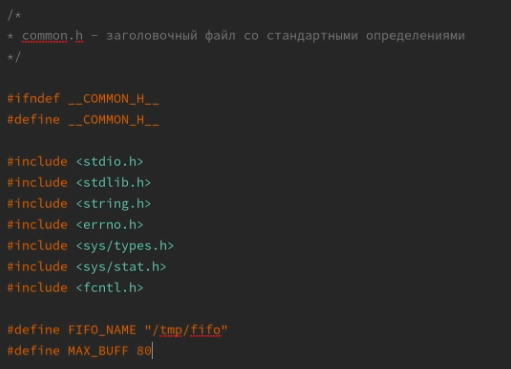


Figure 2: common.h

Файл server.c (рис. [3](#fig:003)).

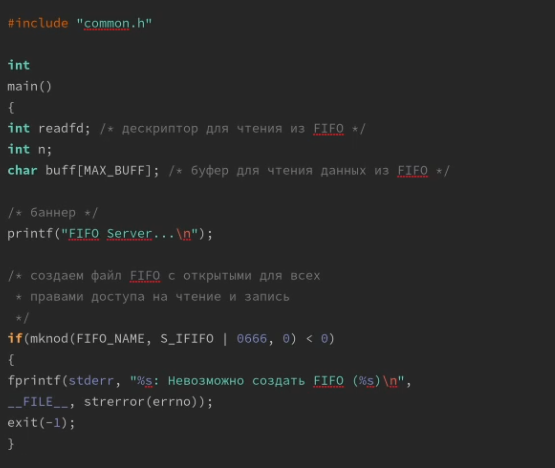


Figure 3: server.c

Файл client.c (рис. [4](#fig:004)).

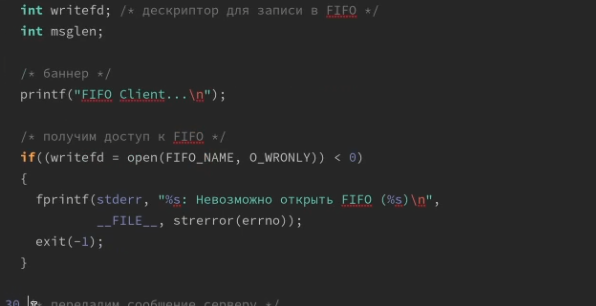


Figure 4: client.c

Makefile (рис. [5](#fig:005)).

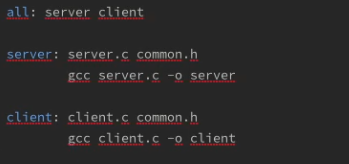


Figure 5: Makefile

Запуск server.c (рис. [6](#fig:006)).

Figure 6: Запуск программы

Figure 6: Запуск программы

Запуск client.c (рис. [7](#fig:007)).

Figure 7: Запуск программы

Figure 7: Запуск программы

# 3 Вывод

Я приобрел практические навыки работы с именованными каналами.

# 4 Контрольные вопросы

1. Именованные каналы (FIFO) - это файлы, находящиеся в файловой системе, которые используются для обмена данными между процессами. Неименованные каналы - это временные файлы, создаваемые с помощью системного вызова pipe, которые используются для передачи данных между процессами, запущенными в рамках одного компьютера.
2. Нет, создание неименованного канала из командной строки невозможно.
3. Да, создание именованного канала из командной строки возможно с помощью утилиты mkfifo.
4. Функция pipe создает неименованный канал и возвращает два файловых дескриптора, один для чтения и один для записи в канал.
5. Функция mkfifo создает именованный канал с заданным именем и правами доступа и возвращает 0 в случае успешного создания и -1 в случае ошибки.
6. При чтении из FIFO меньшего числа байтов, чем находится в канале, процесс будет заблокирован до тех пор, пока не появятся новые данные в канале. При чтении большего числа байтов, процесс получит только те данные, которые есть в канале на данный момент.
7. При записи в FIFO меньшего числа байтов, чем позволяет буфер, данные будут записаны в канал, но процесс записи не будет завершен до тех пор, пока не будет записано достаточное количество данных, чтобы заполнить буфер. При записи большего числа байтов, данные будут записаны в канал, а оставшиеся данные будут ожидать записи в буфере.
8. Два или более процессов могут читать или записывать в канал, но при этом может возникнуть проблема “гонки” (race condition), когда два процесса пытаются одновременно читать или записывать в канал. Для решения этой проблемы необходимо использовать синхронизацию процессов.
9. Функция write используется для записи данных в файл или файлоподобное устройство. Возвращает количество записанных байтов или -1 в случае ошибки. Число 1 в вызове функции в программе server.c (строка 42) означает, что записывается 1 байт.
10. Функция strerror возвращает строку с описанием ошибки, соответствующей заданному коду errno. Она принимает один аргумент - код ошибки errno. Например, strerror(errno) вернет строку, описывающую ошибку, которая произошла в программе.