

Отчёт

ИДЗ №3

Вариант №4

4-5 баллов:

1. Клиент-серверное приложение состоит из 3 программ: сервер, клиент читатель и клиент писатель, код к данным программам находится в папке **4_5grade**.

Сервер получает на вход два аргумента из командной строки: **Ip адрес и номер порта**. Сервер формирует базу данных в виде массива чисел и обрабатывает запросы в зависимости от вида клиента, при запросе читателя сервер отправляет данные из БД, при запросе писателя сервер меняет данные и отправляет их обратно писателю. При изменении базы данных сервер поддерживает ее в непротиворечивом состоянии с помощью встроенной сортировки, что соответствует условию задания.

Клиент читатель получает на вход три аргумента из командной строки: **Ip адрес сервера, номер порта и количество читателей**. Данный клиент посылает заданное количество запросов серверу на чтение данных из базы данных, получает данные и выводит в консоль.

Клиент писатель получает на вход три аргумента из командной строки: **Ip адрес сервера, номер порта и количество писателей**. Данный клиент посылает заданное количество запросов серверу на изменение данных, далее получает ответ от сервера и выводит измененные данные в консоль.

2. Клиенты работают заданное пользователем количество раз или могут быть прерваны из терминала (ctrl + c), тогда сработает обработка сигнала прерывания и произойдет закрытие сокет дескрипторов. Сервер работает, пока не получит сигнал прерывания из терминала (ctrl + c). Данный сигнал обрабатывается с помощью функции **my_handler**, где происходит также закрытие сокет дескриптора.
3. В функции **my_handler** происходит удаление сокет дескриптора.

6-7 баллов:

1. К предыдущему функционалу добавляется клиент Наблюдатель, который выводит состояние сервера, а именно какие запросы обработал сервер, какие данные поменял или прочитал. Клиент Наблюдатель получает на вход два аргумента из командной строки: **Ip адрес и номер порта**. Код данного клиента находится в папке **6_7grade**. Данный клиент подключается к серверу и постоянно опрашивает его об изменениях, если изменений нет, то сервер информирует Наблюдателя об этом, если же изменения появились, то Наблюдатель выводит их в консоль. Наблюдатель работает в онлайн режиме, информация за предыдущие запросы ему недоступна, но это не противоречит условию задания.
2. Наблюдатель работает, пока не получит сигнал прерывания из терминала (ctrl + c). Данный сигнал обрабатывается с помощью функции **my_handler**, где происходит также закрытие сокета дескриптора.

8-9 баллов:

1. К предыдущему функционалу добавляется клиент Наблюдатель, который выводит состояние сервера, а именно какие запросы обработал сервер, какие данные поменял или прочитал. Клиент Наблюдатель получает на вход два аргумента из командной строки: **Ip адрес и номер порта**. Код данного клиента находится в папке **8_9grade**. Данный клиент подключается к серверу и постоянно опрашивает его об изменениях, если изменений нет, то сервер информирует Наблюдателя об этом, если же изменения появились, то Наблюдатель выводит их в консоль. Отличие от предыдущей реализации в том, что к серверу может подключиться несколько Наблюдателей и им будет отображаться одинаковая информация о состоянии сервера на предыдущих запросах. Для этого на сервере было необходимо сформировать историю запросов клиентов писателей и читателей. У каждого Наблюдателя есть свой итератор, который сигнализирует, какие данные он уже получил. Данные наблюдатели могут отключаться и подключаться несколько раз, не изменяя состояние работы сервера.
2. Наблюдатель работает, пока не получит сигнал прерывания из терминала (ctrl + c). Данный сигнал обрабатывается с помощью

функции **my_handler**, где происходит также закрытие сокет дескриптора.