



## ИДЗ 4

Кочетов Тимофей Денисович, БПИ215

Претендую на 9



### Вариант 27. Задача про экзамен.

Преподаватель проводит экзамен у группы студентов. Каждый студент получает свой билет, сообщает его номер и готовит письменный ответ. Подготовив ответ, он передает его преподавателю. Преподаватель просматривает ответ и сообщает студенту оценку. Студент, дождавшись результата, уходит с экзамена. Требуется создать приложение, моделирующее действия преподавателя и студентов, каждый из которых представлен отдельным процессом. Преподаватель — сервер. Каждый студент — отдельный клиент.

## Пояснение

В данной задаче мы имеем следующие сущности и их поведение:

1. **Студенты** - они выбирают билет, готовят ответ и отправляют его на проверку. Затем они ожидают оценку от преподавателя.
2. **Преподаватель** - он получает ответы от студентов, проверяет их и выставляет оценку.

### 3. **Мониторинг** - он (они) выводит информацию о работе всей программы.

Серверное и клиентское приложение моделируют поведение этих сущностей следующим образом:

1. Каждый студент представлен клиентским приложением. Когда клиентское приложение запускается, оно создает соединение с сервером, что аналогично выбору студентом билета. Затем клиент отправляет серверу сообщение (представляющее ответ студента), после чего ожидает ответа от сервера.
2. Преподаватель представлен серверным приложением. Сервер слушает входящие соединения от клиентов. Когда сервер получает соединение от клиента, он запускает новый поток для обработки этого клиента. В этом потоке сервер читает сообщение от клиента, что аналогично просмотру ответа студента. Затем сервер отправляет клиенту оценку, которую он генерирует случайным образом.
3. Мониторинг представлен клиентским приложением. Когда клиентское приложение запускается, оно создает соединение с сервером. Мониторинг выводит все сообщения присылаемые ему от Преподавателя (сервера)

Вся связь между клиентами и сервером осуществляется через сокеты TCP, что обеспечивает надежную доставку сообщений. Каждый клиент и сервер имеют свои собственные процессы, что позволяет им работать независимо друг от друга и одновременно обрабатывать несколько студентов.

Процессы сервера и клиентов могут быть распределены по разным компьютерам в сети, что соответствует ситуации, когда студенты и преподаватель находятся в разных местах, но все еще могут взаимодействовать друг с другом для проведения экзамена.

Также предусмотрена возможность отображения информации о работе приложения с нескольких устройств (несколько мониторингов могут работать одновременно).

Есть возможность подключать и отключать любых клиентов с сохранением работоспособности сервера. После этого можно вновь запускать отключенных клиентов, чтобы приложение в целом могло продолжить свою работу.

## Файлы

- `server.c` - Код серверной части
- `client.c` - Код клиентской части
- `monitor.c` - Код мониторинга
- `server` - Исполняемый файл серверной части
- `client` - Исполняемый файл клиентской части
- `monitor` - Исполняемый файл мониторинга

## Использование

### Компиляция

Чтобы скомпилировать можно использовать **gcc**

```
gcc -o server server.c -lpthread
gcc -o client client.c
gcc -o monitor monitor.c
```

### Запуск

В 1-ом терминале запустить сервер с указанным портом (*например 2024*)

```
./server 2024
```

Во 2-ом терминале запустить клиент с указанным адресом и портом (*например 127.0.0.1 и 2024*)

```
./client 127.0.0.1 2024
```

Во 3-ем терминале запустить мониторинг с указанным адресом и портом (*например 127.0.0.1 и 2024*)

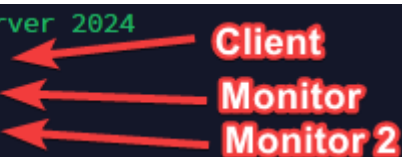
```
./monitor 127.0.0.1 2024
```

## Взаимодействие

В клиенте нужно написать номер билета, в ответ сервер посылает случайно сгенерированную оценку от преподавателя. Мониторинг выводит информацию о проходящем полученную с сервера.

Также можно запустить параллельно сразу несколько мониторингов.

```
root@box-706628:~/OS4_HW# ./server 2024
Connected client IP: 127.0.0.1
Connected client IP: 127.0.0.1
Connected client IP: 127.0.0.1
```



Server

```
root@box-706628:~/OS4_HW# ./client 127.0.0.1 2024
Connected.....
Input ticket number(Q to quit): 123
Message from server: Your grade is 3
Input ticket number(Q to quit): 543
Message from server: Your grade is 1
Input ticket number(Q to quit): Q
```

Client

```
root@box-706628:~/OS4_HW# ./monitor 127.0.0.1 2024
Connected.....
Message from server: Generated grade for student: 3
Message from server: Generated grade for student: 1
```

Monitor

```
root@box-706628:~/OS4_HW# ./monitor 127.0.0.1 2024
Connected.....
Message from server: Generated grade for student: 3
Message from server: Generated grade for student: 1
```

Monitor 2