Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования  
**"Национальный Исследовательский Университет ИТМО"**

Факультет Программной Инженерии и Компьютерных Технологий

**Лабораторная работа №3**

по дисциплине

**«Низкоуровневое программирование»**

Вариант “XML”

Выполнил:

Студент группы P33302

Тюрин Святослав Вячеславович

Преподаватель

Кореньков Юрий Дмитриевич

Санкт Петербург

2023

Задание

На базе данного транспортного формата описать схему протокола обмена информацией и воспользоваться существующей библиотекой по выбору для реализации модуля, обеспечивающего его функционирование. Протокол должен включать представление информации о командах создания, выборки, модификации и удаления данных в соответствии с данной формой, и результатах их выполнения. Используя созданные в результате выполнения заданий модули, разработать в виде консольного приложения две программы: клиентскую и серверную части. Серверная часть – получающая по сети запросы и операции описанного формата и последовательно выполняющая их над файлом данных с помощью модуля из первого задания. Имя фала данных для работы получать с аргументами командной строки, создавать новый в случае его отсутствия. Клиентская часть – в цикле получающая на стандартный ввод текст команд, извлекающая из него информацию о запрашиваемой операции с помощью модуля из второго задания и пересылающая её на сервер с помощью модуля для обмена информацией, получающая ответ и выводящая его в человеко-понятном виде в стандартный вывод.

1 Изучить выбранную библиотеку

a. Библиотека должна обеспечивать сериализацию и десериализацию с валидацией в

соответствии со схемой

b. Предпочтителен выбор библиотек, поддерживающих кодогенерацию на основе схемы

c. Библиотека может поддерживать передачу данных посредством TCP соединения

• Иначе, использовать сетевые сокеты посредством API ОС

d. Библиотека может обеспечивать диспетчеризацию удалённых вызовов

• Иначе, реализовать диспетчеризацию вызовов на основе информации о виде команды

2 На основе существующей библиотеки реализовать модуль, обеспечивающий взаимодействие

a. Описать схему протокола в поддерживаемом библиотекой формате

• Описание должно включать информацию о командах, их аргументах и результатах

• Схема может включать дополнительные сущности (например, для итератора)

b. Подключить библиотеку к проекту и сформировать публичный интерфейс модуля с

использованием встроенных или сгенерированных структур данных используемой библиотеки

• Поддержать установление соединения, отправку команд и получение их результатов

• Поддержать приём входящих соединений, приём команд и отправку их результатов

c. Реализовать публичный интерфейс посредством библиотеки в соответствии с п1

3 Реализовать серверную часть в виде консольного приложения

a. В качестве аргументов командной строки приложение принимает:

• Адрес локальной конечной точки для прослушивания входящих соединений

• Имя файла данных, который необходимо открыть, если он существует, иначе создать

b. Работает с файлом данных посредством модуля из задания 1

c. Принимает входящие соединения и взаимодействует с клиентами посредством модуля из п2

d. Поступающая информация о запрашиваемых операциях преобразуется из структур данных

модуля взаимодействия к структурам данных модуля управления данными и наоборот

4 Реализовать клиентскую часть в виде консольного приложения

a. В качестве аргументов командной строки приложение принимает адрес конечной точки для

подключения

b. Подключается к серверу и взаимодействует с ним посредством модуля из п2

c. Читает со стандартного ввода текст команд и анализирует их посредством модуля из задания 2

d. Преобразует результат разбора команды к структурам данных модуля из п2, передаёт их для

обработки на сервер, возвращаемые результаты выводит в стандартный поток вывода

5 Результаты тестирования представить в виде отчёта, в который включить:

d. В части 3 привести пример сеанса работы разработанных программ

e. В части 4 описать решение, реализованное в соответствии с пп.2-4

f. В часть 5 включить составленную схему п.2а

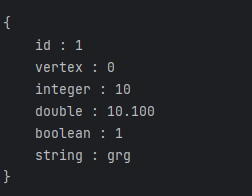
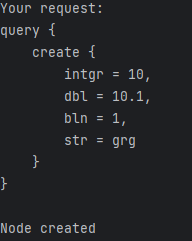
Ход работы

* Клиент получает запрос, парсит его в структуру из прошлой лабораторной request, затем упаковывает эту структуру в XML и отправляет запрос на сервер. Получает ответ и выводит его на экран.
* Сервер в свою очередь получает запрос, определяется с типом запроса, дергает нужную crud функцию, получает результат (опционально) и формирует ответ, затем отправляет.

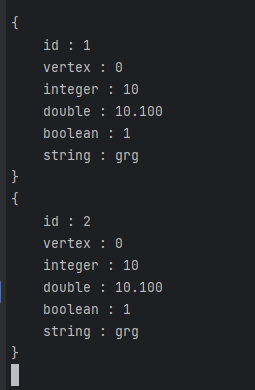
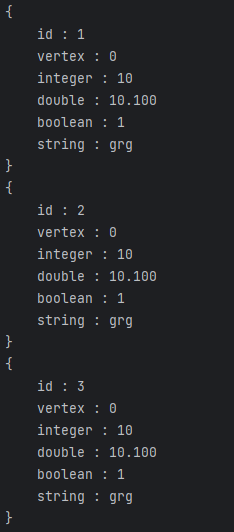
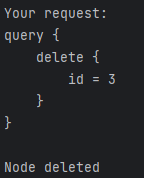
Пример работы программы

Слева окно клиента, справа сервера.

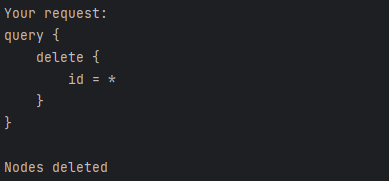
Добавление node



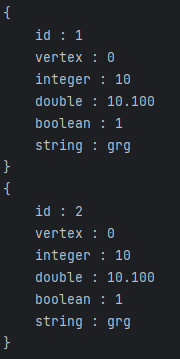
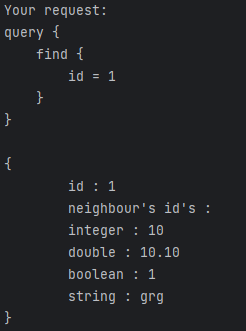
Удаление node по id



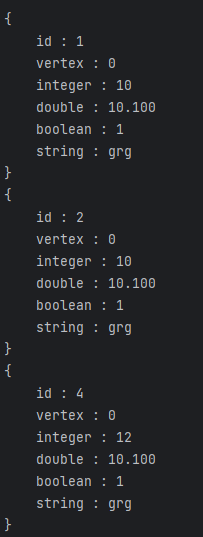
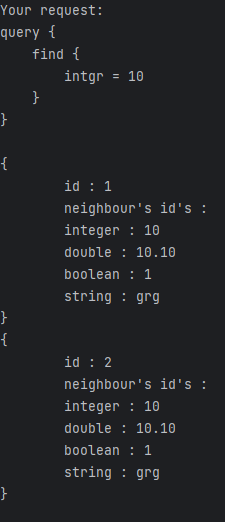
Удаление всех nodes



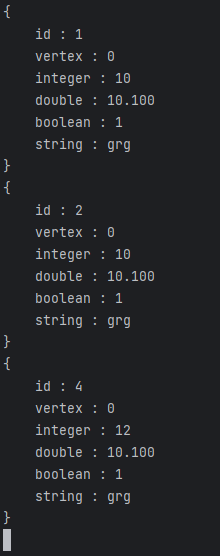
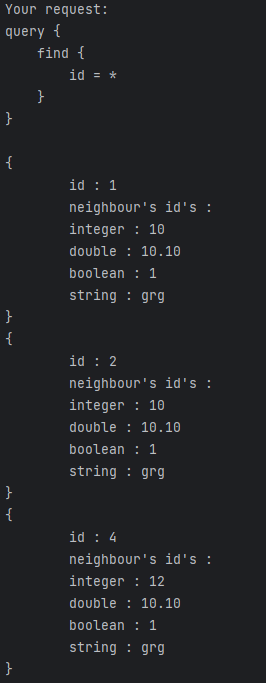
Поиск node по id



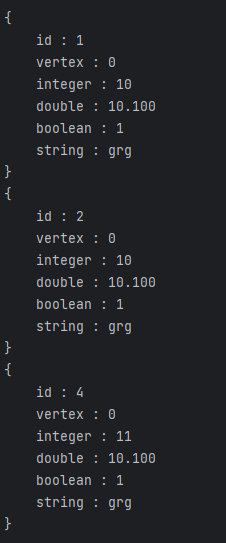
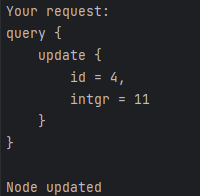
Поиск всех nodes, удовлетворяющих условию



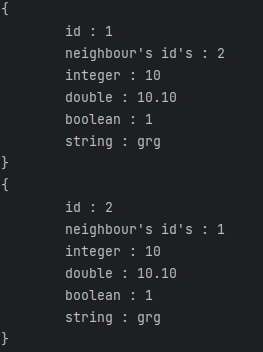
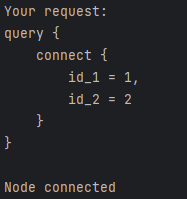
Поиск всех node



Обновление поля node по id

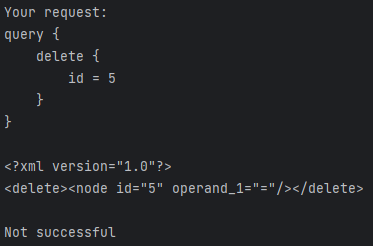


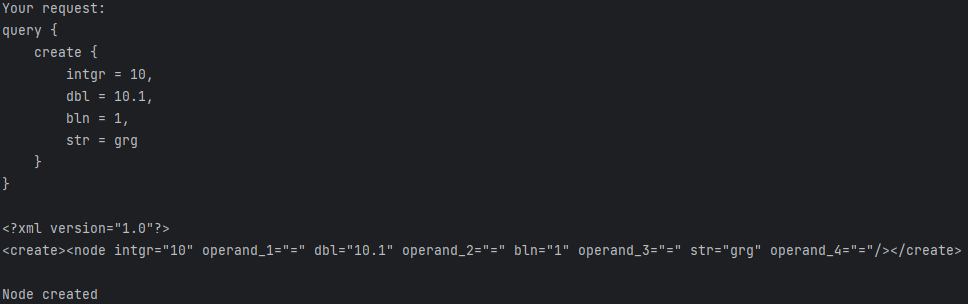
Присоединение двух nodes по id



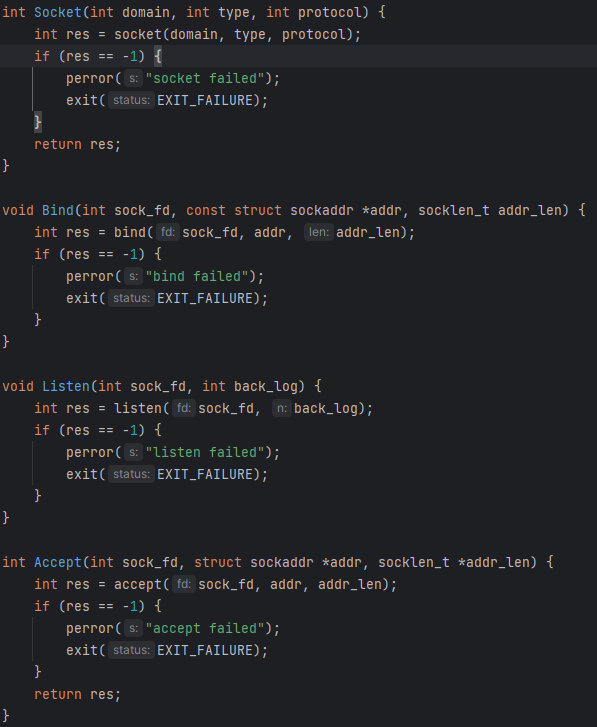
Аспекты реализации

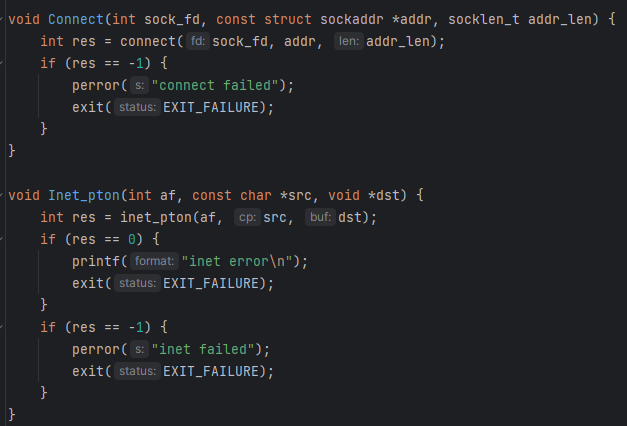
Как выглядит упакованный запрос в XML.

~~~~

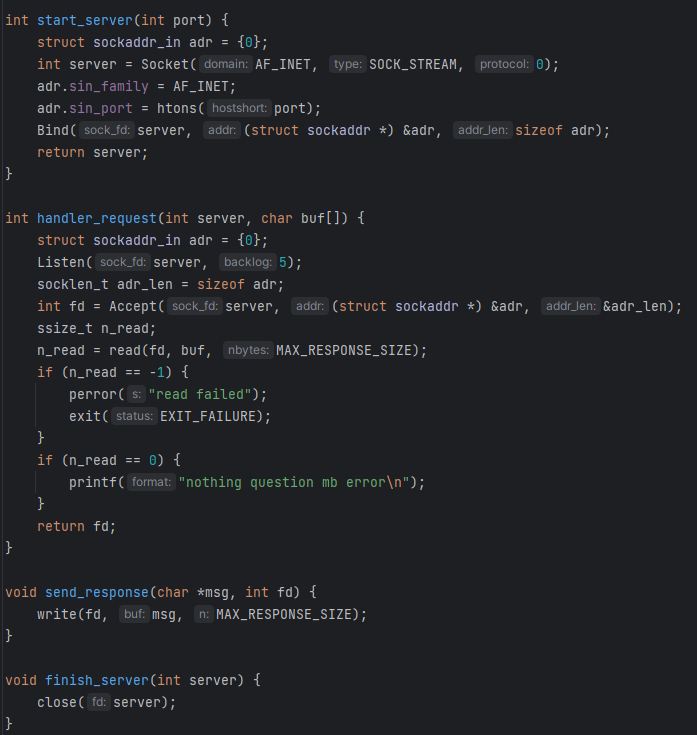
~~~~

Передача по сети происходит через API OC.

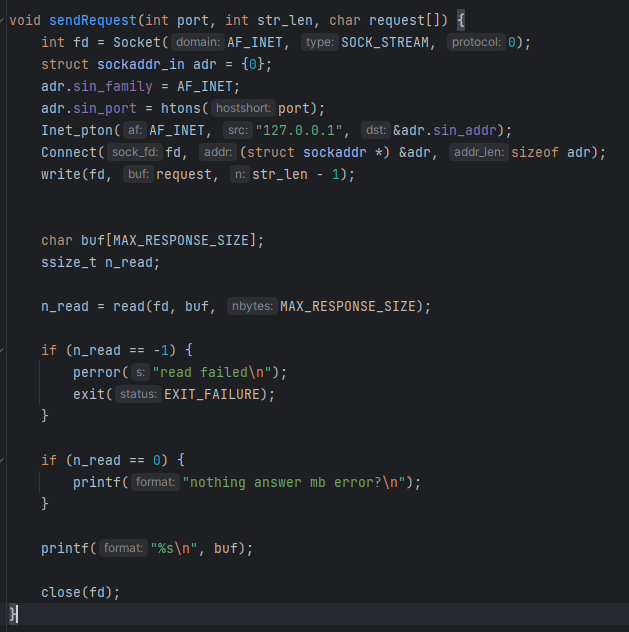




Сервер



Клиент



Пример работы на тестовом наборе со связью, как можно заметить результат идентичный:

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, число

Автоматически созданное описание