

Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и технологий
Кафедра компьютерных систем и программных технологий

Отчет по лабораторной работе

Технологии компьютерных сетей

Работу выполнила:

Белобородова В.Г.

Группа: 43501/1

Преподаватель:

Алексюк А.О.

Санкт-Петербург
2019

1 Индивидуальное задание

1.1 Задание

Разработать распределенную систему, состоящую из приложений клиента и сервера, для распределенного выставления/просмотра курсов валют. Информационная система должна обеспечивать параллельную работу нескольких клиентов.

1.2 Основные возможности

Серверное приложение должно реализовывать следующие функции:

1. Прослушивание определенного порта
2. Обработка запросов на подключение по этому порту от клиентов
3. Поддержка одновременной работы нескольких клиентов через механизм нитей
4. Добавление новой валюты (кода валюты)
5. Удаление валюты
6. Добавление курса конкретной валюты
7. Выдача пользователю списка имеющихся валют с текущими курсами и абсолютными/относительными приращениями к предыдущим значениям
8. Выдача пользователю истории курса конкретной валюты

Клиентское приложение должно реализовывать следующие функции:

1. Возможность параллельной работы нескольких клиентов с одного или нескольких IP-адресов
2. Установление соединения с сервером
3. Разрыв соединения
4. Обработка ситуации отключения сервером
5. Получение и вывод списка валют с котировками/изменениями
6. Передача команды на добавление валюты
7. Передача команды на удаление валюты
8. Передача команды на добавление курса валюты
9. Получение и вывод истории котировок валюты

1.3 Настройка приложений

Разработанное клиентское приложение должно предоставлять пользователю возможность задания IP-адреса или доменного имени сервера, а также номера порта сервера.

1.4 Тестирование

Для тестирования запускается сервер системы котировок валют и несколько клиентов. В процессе тестирования проверяются основные функциональные возможности разработанной системы.

2 Прикладной протокол

Команды протокола имеют следующий вид:

Таблица 1: Команды прикладного протокола

Команда	Описание	Ответ	Пояснение к ответу
l	Получить список валют	(Таблица с пронумерованными названиями валют, текущим курсом и изменением относительно предыдущих значений)	–
a имя_валюты	Добавить валюту с названием имя_валюты	Success / Error	Добавление валюты будет неуспешным, если на сервере сейчас существуют максимально возможное количество валют
d номер_валюты	Удалить валюту под номером номер_валюты	Error. Please enter currency number / Error. Wrong currency number / Error. Currency does not exist / Success	Удаление будет неуспешным в случаях если: после d не указан номер валюты, указан неверный (отрицательный или превышающий максимальный) номер валюты, указан номер не действующей сейчас валюты
c номер_валюты значение_валюты	Обновить значение валюты под номером номер_валюты на новое значение значение_валюты	Error. Please enter currency number / Error. Wrong currency number / Error. Currency does not exist / Error. Please enter currency value / Error. Wrong value format / Error. Same value / Success	Операция будет неуспешной, если нарушен формат команды/параметров, а также если пользователь пытается задать валюте то же значение, что у неё сейчас

s номер_валюты	Посмотреть историю изменений значения валюты под номером номер_валюты	Error. Please enter currency number / Error. Wrong currency number / Error. Currency does not exist / (История курса валюты)	Сервер вернёт сообщение об ошибке, если нарушен формат команды/параметра или запрашиваемая валюта не существует
q	Клиент уведомляет сервер о завершении сеанса (сообщение отправляется клиентским приложением, пользователь может не знать о существовании такой команды)	–	Сервер разрывает соединение с клиентом и освобождает ресурсы для нового клиента
(Любая другая команда)	Команда, не соответствующая ни одному из шаблонов	Unknown command. Try again	–

3 Реализация сервера

3.1 Использование сервера

Для запуска сервера достаточно запустить исполняемый файл, параметры для запуска приложения не требуются. Во время работы приложения сервер можно отключить путём ввода символа 'q'. Для взаимодействия с информационной частью сервера (таблицами с валютами) необходимо подключиться к серверу с клиентского приложения.

3.2 Алгоритм работы сервера

Алгоритм запуска сервера описывается следующими шагами:

1. Создание tcp сервер-сокета с заданными параметрами
2. Запуск контрольного потока (с этого момента сервер можно отключить командой 'q')
3. Прослушивание порта и выделение каждому новому клиенту потока, в котором его будут обслуживать (этот этап не ограничен по времени)
4. Получение команды выключения сервера 'q'
5. Закрытие всех активных соединений с клиентами
6. Вывод сообщения об успешном завершении работы сервера

Стоит отметить, что tcp удобен тем, что соединение между клиентом и сервером не обязательно устанавливать каждый раз по новой для каждого запроса. Однажды установленное

соединение остаётся активным, пока клиент не отправит уведомление об отключении или пока одна из сторон не обнаружит разрыв соединения.

Алгоритм обработки запросов довольно массивен, но примитивен, поэтому он не заслуживает подробного описания. Стоит лишь отметить, что запрос 'q' (запрос отключения, пришедший от клиента) является особым, и его обработка производится вне общей функции обработки запросов. Это связано с тем, что реакцией на 'q' должен быть не текстовый ответ, а ряд действий по отключению клиента, поэтому обработка этой команды не вписывается в общую функцию.

3.3 Представление информации о валютах

Для хранения информации о валютах используются следующие структура и массив:

```
1 #define maxRecords 5
2 #define storyLength 3
3 #define nameLength 3
4
5 struct currencyData
6 {
7     char name[nameLength];
8     int value;
9     int story[storyLength];
10    int actual;
11 };
12 struct currencyData mainMass[maxRecords];
```

Поле "actual"— поле с логическим значением 0/1. Значение в этом поле показывает, используется ли сейчас данный элемент массива mainMass. Такой подход необходим для повторного использования одних и тех же элементов массива при многократном добавлении и удалении валют, так как фактически при удалении валюты элемент массива не удаляется.

3.4 Исходный код

Исходный код находится в репозитории по ссылке: <server/main.c> .

4 Реализация клиента

4.1 Использование клиента

При запуске приложения-клиента необходимо указать IP-адрес и номер порта сервера, например:

```
./client_linux 127.0.0.1 5001
```

Затем пользователь может вводить команды согласно прикладному протоколу. Завершить сеанс работы с сервером можно вводом команды 'q'.

4.2 Алгоритм работы клиента

Алгоритм запуска клиента описывается следующими шагами:

1. Считывание параметров (IP-адрес, номер порта)

2. Установление соединения с сервером
3. Считывание команды пользователя, отправка запроса на сервер, отображение ответа сервера (количество повторений данного шага не ограничено)
4. Получение команды завершения сеанса
5. Разрыв соединения с сервером
6. Вывод сообщения об успешном завершении работы приложения

Клиентское приложение после отправки сообщения на сервер просматривает первый символ отправленного сообщения. Если этот символ – 'q', то происходит выход из цикла обмена с сервером. Если же первый символ не q, то ожидается ответ от сервера.

4.3 Исходный код

Исходный код находится в репозитории по ссылке: `<client/main.c>` .

5 Тестирование

5.1 Один клиент

При помощи одного клиента проверена корректность работы основных функций сервера. Лог с консоли клиента представлен ниже:

```
1 $ ./client_linux 127.0.0.1 5001
2 Please enter the command: a CU1
3 Success
4 —End of the response—
5 Please enter the command: a CU2
6 Success
7 —End of the response—
8 Please enter the command: l
9
10 1) CU1: 0
11 2) CU2: 0
12
13 —End of the response—
14 Please enter the command: c 2 30
15 Success
16 —End of the response—
17 Please enter the command: l
18
19 1) CU1: 0
20 2) CU2: 30
21
22 —End of the response—
23 Please enter the command: c 2 40
24 Success
25 —End of the response—
26 Please enter the command: l
```

27
28 1) CU1: 0
29 2) CU2: 40 (+10)
30
31 —End of the response—
32 Please enter the command: c 2 20
33 Success
34 —End of the response—
35 Please enter the command: l
36
37 1) CU1: 0
38 2) CU2: 20 (−20)
39
40 —End of the response—
41 Please enter the command: s 2
42 CU2: 20 − 40 − 30 − 0
43 —End of the response—
44 Please enter the command: d 1
45 Success
46 —End of the response—
47 Please enter the command: l
48
49 2) CU2: 20 (−20)
50
51 —End of the response—
52 Please enter the command: a CU3
53 Success
54 —End of the response—
55 Please enter the command: l
56
57 1) CU3: 0
58 2) CU2: 20 (−20)
59
60 —End of the response—
61 Please enter the command: s 4
62 Error. Currency does not exist
63 —End of the response—
64 Please enter the command: d 8
65 Error. Wrong currency number
66 —End of the response—
67 Please enter the command: c 2 qwe
68 Error. Wrong value format
69 —End of the response—
70 Please enter the command: pri vet
71 Unknown command. Try again
72 —End of the response—
73 Please enter the command: c 2 20
74 Error. Same value
75 —End of the response—

В ходе данного теста подтверждено, что сервер корректно обрабатывает как верные, так и

неверные запросы и присылает ответы в удобном для чтения виде.

5.2 Два клиента

Производилась отправка команд с двух одновременно подключенных к серверу клиентов. После изменения одним клиентом состояния таблицы валют, второй клиент при запросе таблицы получал её обновлённую версию. Скриншот данного эксперимента представлен по ссылке: <скриншот> .

6 Вывод

Разработано клиент-серверное приложение выставления курса валют на основе протокола ТСР. Протокол ТСР в данном случае удобен тем, что позволяет открыть соединение один раз и затем держать его открытым на всё время обмена с клиентом. Приложение приближено к реальному тем, что позволяет одновременно работать с сервером несколькими клиентами.