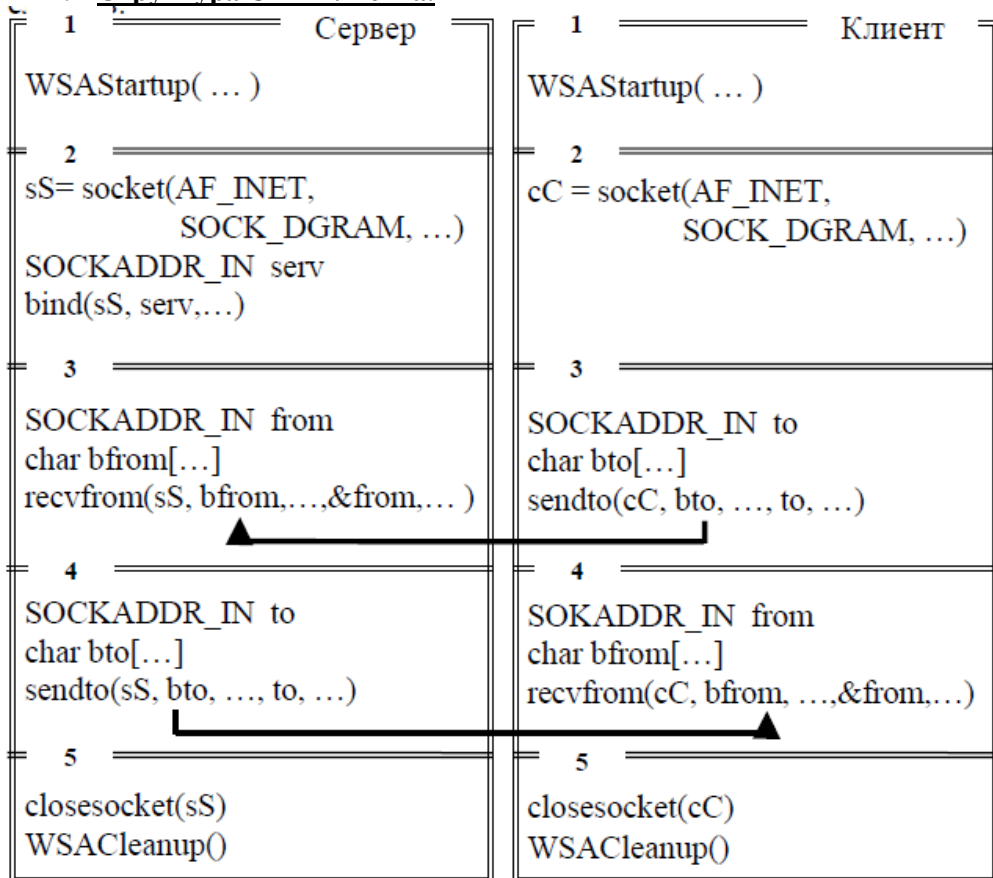


1. Структура UDP-сервера.

2. Структура UDP-клиента.



Первые блоки обеих программ одинаковые и предназначены для инициализации библиотеки WS2_32.dll. Второй блок программы-сервера создает сокет (функция socket) и устанавливает его параметры. Следует обратить внимание на параметр SOCK_DGRAM функции socket, указывающий на тип сокета (в данном случае – сокет, ориентированный на сообщения). Для установки параметров сокета используется функция bind. При этом говорят, что сокеты связывают параметрами. Для хранения параметров сокета в Winsock2 предусмотрена специальная структура SOCKADDR_IN. перед выполнением функции bind, которая использует эту структуру в качестве параметра, необходимо ее заполнить данными. В SOCKADDR_IN хранится IP-адрес и номер порта сервера.

В третьем блоке программы сервера выполняется функция recvfrom, которая переводит программу сервера в состояние ожидания до поступления сообщения от программы клиента (функция sendto). Функция recvfrom тоже использует структуру SOCKADDR_IN. В нее автоматически помещаются параметры сокета клиента после приема от него сообщения. Данные поступают

в буфер, который обеспечивает принимающая сторона (на рисунке символичный массив `bfrom`). Следует отметить, что в качестве параметра функции `recvfrom` используется связанный сокет и именно через него осуществляется передача данных.

Четвертый блок программы сервера предназначен для пересылки данных клиенту. Процесс осуществляется с помощью функции `sendto`. В качестве параметров `sendto` использует структуру `SOCKADDR_IN` с параметрами сокета принимающей стороны (в данном случае клиента) и заполненный буфер с данными.

Пятые блоки программ сервера и клиента одинаковые и предназначены для закрытия сокета и завершения работы с библиотекой `WS2_32.dll`.

Всем блокам программы клиента, кроме второго, есть аналог в программе сервера. Второй блок по сравнению с сервером не использует команду `bind`. Здесь проявляется основное отличие между сервером и клиентом. Если сервер должен использовать однозначно определенные параметры (IP-адрес и номер порта), то для клиента это не обязательно – ему Windows выделяет эфемерный порт. Поскольку инициатором связи является клиент, то он должен точно «знать» параметры сокета сервера, а свои параметры клиент получит от Windows и сообщит их вместе с переданным пакетом серверу.

3. Отличие UDP-сервера от TCP-сервера.

4. Отличие UDP-клиента от TCP-клиента.

5. Что такое канал связи с точки зрения TCP?

Канал связи (или соединение) создается между двумя сокетами, ориентированными на поток. На стороне сервера это должен быть связанный (функция `bind`) и переключенный в режим прослушивания (функция `listen`) сокет. На стороне клиента должен быть создан дескриптор ориентированного на поток сокета (функция `socket`).

Канал связи создается в результате взаимодействия функция `accept` (на стороне сервера) и `connect` (на стороне клиента). Алгоритм взаимодействия этих функция зависит от установленного режима ввода-вывода для участвующих в создании каналов сокетов.

6. Что значит: обмен без соединения (ориентированный на сообщения)?

Это такой обмен данными (сообщениями), особенностью которого является то, что протоколом, который обеспечивает такой обмен, не гарантируется доставка и правильная последовательность приема отправленных сообщений. Весь контроль надежности доставки сообщений возлагается на разработчика приложения. В связи с этим, обмен данными с помощью сообщений используется в основном для широковещательных сообщений или

для пересылки коротких сообщений, последовательность получения которых не имеет значения.

7. Что значит UDP ненадежный протокол?

Ненадежный протокол – это такой протокол, который обладает следующими свойствами:

- отсутствует механическое обеспечение надежности: пакеты не упорядочиваются, и их прием не подтверждается;
- отсутствует гарантия доставки;
- отсутствует обработка соединений
- невозможно вычислить контрольную сумму для пакета данных
- отсутствие буферизации
- нельзя разбивать сообщения на несколько пакетов

8. В каких случаях следует выбирать UDP-протокол для обмена данными?

Обмен данными с помощью сообщений используется в основном для широковещательных сообщений или для пересылки коротких сообщений, последовательность получения которых не имеет значения.

9. Что значит: UDP не буферизирует соединения?

Обрабатывает только один пакет, который пришел. Пока не обработает этот пакет, другие пакеты не обрабатываются, т.к. нет буфера памяти.

10. Можно ли одновременно задействовать один номер порта для двух серверов: TCP и TCP, UDP и UDP, TCP и UDP? если возможно, то продемонстрируйте.

- TCP и TCP – запустить можно, но работать (передавать данные) одновременно он не сможет.
- UDP и UDP – нельзя (сразу же срабатывает ошибка).
- TCP и UDP – запустить можно, но работать (передавать данные) одновременно он не сможет.
-

11. Определите порт UDP-сервера с помощью утилиты NETSTAT. 0.0.0.0:2000 (порт 2000)

12. Пр продемонстрируйте потерю сообщений UDP-сервером. Используйте функцию Sleep.