

Лабораторная работа №7

Передача данных между процессами. Часть III

(2 часа)

Содержание: структура клиентского и серверного сетевого приложений Windows Sockets 2 на базе дейтаграммных сокетов.

Цель: изучить основы построения сетевых приложений на базе Windows Sockets 2, получить навыки использования основных функции API WinSock2 при разработке UDP-сервера и UDP-клиента.

Передача данных при помощи дейтаграмм протокола UDP подобна передаче данных при помощи mailslot'ов. Точно так же не требуется установка соединения, данные передаются сообщениями, получателей может быть больше одного, а доставка данных не гарантируется, также как и порядок доставляемых сообщений.

Работа по передаче данных с использованием протокола UDP также как и с использованием протокола TCP начинается с создания сокета функцией `socket`. При этом тип сокета задаётся как `SOCK_DGRAM`.

После создания сокета можно сразу переходить к приёму / передаче данных. В случае протокола UDP для этой цели используются функции `sendto` и `recvfrom`. Эти функции принимают такие же аргументы, как и рассмотренные в предыдущей работе `send` / `recv`, которые дополняются ещё двумя параметрами – адресом структуры `SOCKADDR` и её размером для сохранения адресной информации другой стороны сетевого взаимодействия.

```
int sendto (
    SOCKET s,
    LPSTR lpBuffer,
    int nBufferLen,
    int nFlags,
    LPSOCKADDR lpAddr,
    int nAddrLen);
```

Здесь `lpAddr` указывает на адресную структуру, через которую задаётся IP-адрес и номер порта конкретной системы, которой адресуется информация или указывается, что дейтаграмма должна рассылаться всем.

Одно из отличий использования протокола UDP от TCP в том, что UDP позволяет передавать информацию одновременно более чем одному получателю, то есть широковещательно. Для этого необходимо выполнить два действия. Во-первых, после создания сокета установить режим его работы в широковещательный (в сетевых технологиях используется термин **broadcast**). Например, для сокета `Socket` это можно сделать так:

```
int AllowBroadcast = 1;
if (setsockopt(Socket, SOL_SOCKET, SO_BROADCAST,
(char*)&AllowBroadcast, sizeof(AllowBroadcast)) < 0) {
```

```
cout << "Ошибка перевода сокета в режим широковещания: "  
<< WSAGetLastError() << endl;  
closesocket(Socket);  
system("pause");  
return 1;  
}
```

Во-вторых, в поле адреса структуры `sockaddr_in` нужно задать значение `INADDR_BROADCAST` или IP-адрес `255.255.255.255` (другие варианты адресов будут рассмотрены в курсе компьютерных сетей).

Параметры функции `recvfrom` аналогичны функции `sendto`, за исключением того, что в ней указывается адресная информация системы от которой принимаются данные.

Задание. Добавьте в задание предыдущей лабораторной работы возможность автоматического обнаружения сервера, основанную на протоколе UDP.