编程步骤

1、server端

（1）加载套接字库，创建套接字（WSAStartup()/socket()）;

#include<winsock.h>

#pragma comment(lib,"ws2\_32.lib")

void initialization();

int main()

{

//创建套接字

s\_server = socket(AF\_INET, SOCK\_STREAM, 0);

}

void initialization() {

//初始化套接字库

WORD w\_req = MAKEWORD(2, 2);//版本号

WSADATA wsadata;

int err;

err = WSAStartup(w\_req, &wsadata);

if (err != 0) {

cout << "初始化套接字库失败！" << endl;

}

else {

cout << "初始化套接字库成功！" << endl;

}

//检测版本号

if (LOBYTE(wsadata.wVersion) != 2 || HIBYTE(wsadata.wHighVersion) != 2) {

cout << "套接字库版本号不符！" << endl;

WSACleanup();

}

else {

cout << "套接字库版本正确！" << endl;

}

//填充服务端地址信息

}

 （2）绑定套接字到一个IP地址和一个端口上（bind()）;

server\_addr.sin\_addr.S\_un.S\_addr = htonl(INADDR\_ANY);

server\_addr.sin\_port = htons(5010);

   （3）将套接字设置为监听模式等待连接请求（listen()）;

//设置套接字为监听状态

if (listen(s\_server, SOMAXCONN) < 0)

{

cout << "设置监听状态失败！" << endl;

WSACleanup();

}

else

{

cout << "设置监听状态成功！" << endl;

}

cout << "服务端正在监听连接，请稍候...." << endl;

   （4）请求到来后，接受连接请求，返回一个新的对应于此次连接的套接字（accept()）;

//接受连接请求

len = sizeof(SOCKADDR);

s\_accept = accept(s\_server, (SOCKADDR \*)&accept\_addr, &len);

if (s\_accept == SOCKET\_ERROR)

{

cout << "连接失败！" << endl;

WSACleanup();

return 0;

}

cout << "连接建立，准备接受数据" << endl;

    （5）用返回的套接字和客户端进行通信（send()/recv()）;

//接收数据

while (1)

{

recv\_len = recv(s\_accept, recv\_buf, 100, 0);

if (recv\_len < 0)

{

cout << "接受失败！" << endl;

break;

}

else

{

cout << "客户端信息:" << recv\_buf << endl;

}

cout << "请输入回复信息:";

cin >> send\_buf;

send\_len = send(s\_accept, send\_buf, 100, 0);

if (send\_len < 0)

{

cout << "发送失败！" << endl;

break;

}

}

    (6)返回，等待另一个连接请求；

  （7）关闭套接字，关闭加载的套接字库（closesocket()/WSACleanup());

//关闭套接字

closesocket(s\_server);

closesocket(s\_accept);

//释放DLL资源

WSACleanup();

return 0;

     2、Client端

    （1）加载套接字库，创建套接字（WSAStartup()/socket）;

#include<winsock.h>

#pragma comment(lib,"ws2\_32.lib")

void initialization();

int main()

{

//创建套接字

s\_server = socket(AF\_INET, SOCK\_STREAM, 0);

}

void initialization() {

//初始化套接字库

WORD w\_req = MAKEWORD(2, 2);//版本号

WSADATA wsadata;

int err;

err = WSAStartup(w\_req, &wsadata);

if (err != 0) {

cout << "初始化套接字库失败！" << endl;

}

else {

cout << "初始化套接字库成功！" << endl;

}

//检测版本号

if (LOBYTE(wsadata.wVersion) != 2 || HIBYTE(wsadata.wHighVersion) != 2) {

cout << "套接字库版本号不符！" << endl;

WSACleanup();

}

else {

cout << "套接字库版本正确！" << endl;

}

//填充服务端地址信息

}

   (2)向服务器发出连接请求（connect()）;

if (connect(s\_server, (SOCKADDR \*)&server\_addr, sizeof(SOCKADDR)) == SOCKET\_ERROR) {

cout << "服务器连接失败！" << endl;

WSACleanup();

}

else {

cout << "服务器连接成功！" << endl;

}

  (3)和服务器进行通信（send()/recv()）;

//发送,接收数据

while (1) {

cout << "请输入发送信息:";

cin >> send\_buf;

send\_len = send(s\_server, send\_buf, 100, 0);

if (send\_len < 0) {

cout << "发送失败！" << endl;

break;

}

recv\_len = recv(s\_server, recv\_buf, 100, 0);

if (recv\_len < 0) {

cout << "接受失败！" << endl;

break;

}

else {

cout << "服务端信息:" << recv\_buf << endl;

}

}

   (4)关闭套接字，关闭加载的套接字库（closesocket()/WSACleanup()）

//关闭套接字

closesocket(s\_server);

//释放DLL资源

WSACleanup();