C++：cs106L记录

Convert 转换 loop 循环 short hand 速记 clunky.笨重 a silver-bullet solution 精妙解决方案 robust 健壮的

ostringstream oss("Tea"); //非标准，应该没有初始值

oss<<“str”

istringstream iss(str); //用str初始化

iss>>str>>str2

输出前cout<<fixed精度修复

Setw boolalpha

* **setprecision(n)**：设置浮点数的精度，即小数位数为 **n**。
* **setw(n)**：设置输出字段的宽度为 **n** 个字符，用于对齐输出。
* **setbase(base)**：设置整数的输出进制，例如 **setbase(16)** 可以将整数以十六进制形式输出。
* **setiosflags(flag)**：设置指定的输出流标志，例如 **setiosflags(ios::left)** 可以设置左对齐输出。
* **resetiosflags(flag)**：重置指定的输出流标志，将其恢复为默认值。
* **setw(int)** 和 **setfill(char)** 的组合：用于设置字段宽度和填充字符。

cout << '[' << left << setw(10) << "Hello!" << ']' << endl; // [ Hello!]

cout << '[' << right << setw(10) << "Hello!" << ']' << endl; // [Hello! ]

cout << setfill('0') << setw(8) << 1000 << endl; // Prints 00001000

cout << setw(8) << 1000 << endl; // Prints 00001000 because of last setfill

* **hex**：将后续的整数输出为十六进制形式。
* **dec**：将后续的整数输出为十进制形式（默认形式）。
* **oct**：将后续的整数输出为八进制形式。
* 在输入流中使用 **ws** 可以忽略前导和尾随的空白字符，直接读取有效数据。这在处理输入时特别有用，因为它可以帮助你跳过不需要的空白字符。
* while(true) {
* int intValue;
* double doubleValue;
* input >> intValue >> doubleValue;
* if(input.fail()) break;
* cout << setw(COLUMN\_WIDTH) << (rowNumber + 1) << " | ";
* cout << setw(COLUMN\_WIDTH) << intValue << " | ";
* cout << setw(COLUMN\_WIDTH) << doubleValue << endl;
* rowNumber++;
* }

int intValue;

double doubleValue;

while(input >> intValue >> doubleValue) {

/\* ... process values here ... \*/

}

getline(cin, myStr);

cin.ignore(std::numeric\_limits<std::streamsize>::max(), '\n'); //清除缓存区的’\n’

端口号由操作系统分配

Udp协议 0.0.0.0表示监听所有可能的网络ip

Netstat -ano

端口号 20 和 21：FTP (文件传输协议)

端口号 22：SSH (安全外壳协议)

端口号 25：SMTP (简单邮件传输协议)

端口号 53：DNS (域名系统)

端口号 80：HTTP (超文本传输协议)

端口号 443：HTTPS (安全超文本传输协议)

端口号 110：POP3 (邮局协议版本3)

端口号 143：IMAP (Internet消息访问协议)

端口号 3389：RDP (远程桌面协议)

端口号 3306：MySQL数据库服务器

端口号 5432：PostgreSQL数据库服务器

客户端一种

服务端两种：接受请求和负责传输数据的套接字

TCP（Transmission Control Protocol）是一种传输层协议，

TCP（Transmission Control Protocol）是一种传输层协议，它建立在网络层（如IPv4）之上，提供可靠的、面向连接的数据传输。

typedef struct WSADATA {

WORD wVersion;

WORD wHighVersion;

char szDescription[WSADESCRIPTION\_LEN+1];

char szSystemStatus[WSASYSSTATUS\_LEN+1];

unsigned short iMaxSockets;

unsigned short iMaxUdpDg;

char FAR \*lpVendorInfo;

} WSADATA;