1. C++中的输入输出流

一、文本文件的输入与输出

1. 读文件：

引入头文件<fstream>

创建类ofstream来管理写出

创建类ifstream来管理读入

这些类中需要设置open成员函数来与磁盘文件位置相关联

为了健壮性考虑，也可以在打开文件之后加一层判断语句（is\_open）来 检查文件是否正常打开

1. 写文件：
   1. 引入头文件fstream

fstream头文件定义了用于文件输入的类ifstream和文件输出的类ofstream

* 1. 输出操作

1)创建一个ofstream对象来管理输出流

2)将该对象与文件关联起来

3)调用write函数向文件写数据；或使用cout方式使用该对象，唯一区别是输出将进入文件，而不是屏幕

4)关闭到文件的连接

//文本文件的输出也可以使用标准插入符<<

1. 写文件操作示例

class ofstream{

public:

ofstream( char \*pFileName,

int mode=ios::out,

int prof=filebuf::openprot );

void open ( char\*, int );

void close ( );

ofstream & write (char\*, int );

int eof();

……

}

#include <iostream.h>

#include <fstream.h>

void main( )

{ char a[6]="absyw";

ofstream of;

of.open("d:\\t1.doc"); //进行文档关联

of. write((char \*) &a,sizeof a) ;

of. close ( );

}

1. 读文件操作示例

① while( (ch = cin.get() ) ! = EOF )

cout.put( ch);

}

//EOF可用于判断是否输入结束

②getline ( char\*,int, char ):

从输入流中读取多个数据，并且允许指定输入

终止字符（默认值是换行符） 。

③class ifstream{

public:

ifstream( char \*pFileName,

int mode=ios::out,

int prof=filebuf::openprot );

void open ( char\*, int );

void close ( );

ifstream & read (char\*, int );

int eof();

……

}

1. 二进制文件的输入与输出

仅需要一个fstream类来管理输入与输出

Fstream iof;

iof.open(“d:\\tmp.dat”, ios::out|ios::binary);

---->>代码风格：以封装性为优，尽量少地使用全局函数

1. C++中的文件定位

Explain：文件定位的功能是在文件中确定需要进行修改的位置

1. Seekp 管理输出（put）
2. Seekg 管理输入（get）
3. 字符串流操作

#include<sstream>

1. 概述：字符串输入输出流是直接对内存操作，针对string而非是文件流或标准流

它将各种数据类型与字符串互相流动

但它并不流向监视器（屏幕）或磁盘文件，而是直接对内存操作

1. 字符串输出流（ostringstream）

可用于构造（合并）字符串，如将数值直接转为字符串

1. 字符串输入流（istringstream）

可用于输入（拆分）字符串，如将字符串转为其他类型

Attention:选择i/o stream的“输入/输出”是相对于程序而言的

1. 小结

C++中的输入输出包括

1. 标准输入输出（cin,cout）
2. 文件输入输出（ifstream,ofstream,read,write）
3. 字符串输入输出（ostringstream,istringstream）