

计算机等级考试二级 C 各章内容摘要：第8章 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/237/2021_2022__E8_AE_A1_E7_AE_97_E6_9C_BA_E7_c97_237728.htm 第8章 运算符重载

【考点一】运算符函数与运算符重载 运算符重载是计算机语言固有多态性的体现，是构成计算机语言的基础之一。C++把重载的运算符视为特殊的函数，称为运算符函数。运算符重载就是函数重载的一种特殊情况。像对待一般重载函数一样，编译系统能够依据使用运算符的不同环境，即参数（操作数）的数量或类型的差异，区分同一运算符的不同含义。"运算符重载"是针对C++中原有运算符进行的，不可能通过重载创造出新的运算符。除了`、`、`.*`、`->*`、`::`、`?:`这五个运算符外，其他运算符都可以重载。由于很多符号是一元运算符和二元运算符公用的，为了避免含混，不得为重载的运算符函数设置默认值，调用时也就不得省略实参。除了`new`和`delete`这两个较为特殊运算符以外，任何运算符如果作为成员函数重载时不得重载为静态函数。`=`、`[]`、`()`、`->`以及所有的类型转换运算符只能作为成员函数重载，而且不能是针对枚举类型操作数的重载。运算符函数的函数名是由运算符前加关键字`operator`构成的，在声明运算符或调用运算符时都可以用这个名称。 **【考点二】典范运算符的重载**

1.关于分数类`fraction`的声明和定义包含在头文件`fraction.h`和程序文件`fraction.cpp`中。一个标准的用`fraction`表示的分数须满足以下复印件：
分母永远为正，分数和符号用分子表示；
分子分母互质，即总表示为最简分数

。`fraction`通过两个私有数据成员`num`和`den`分别保存分子和分

母，并在必要时调用standardize函数进行标准化处理，以使num和den的值满足标准分数的条件。gcd是求两个整数的最大公约数的函数，standardize在化简分数时要调用它。

2.重载取负运算符"-"

因为fraction用分子的符号代表整个分数的符号，因此所谓"取负"只需对分子num取负就可以了。由于取负运算符"- "是一元运算符，当作为成员函数重载时参数表中没有参数，那个唯一的操作数以this指针的形式隐藏在参数表中。为此，只需要在fraction.h的类声明中增加：`fraction poerator -()const { return fraction(-num, den); }`就可以了。由于在类声明中直接给出了完整定义，因此是一个inline函数。"- "是一个典型的一元运算符，除++、--外的其他一元运算符的重载都可以参考这里描述的方法。

3.重载加法运算符"+"

"+"是一个二元运算符，因此作为成员函数重载时参数表中只有一个参数，对应于第二操作数，而第一操作数就是对象本身，仅以this指针的形式隐藏在参灵敏表中。"+"是一个典型的二元运算符，除赋值类运算符外的其他二元运算符的重载都可以参考这里描述的方法。

4.重载增1运算符"++"

"++"既可以是前缀运算符（前增1），又可以是后缀运算符（后增1）。为了区分这两种情况，重载这两个运算符时必须在格式上有所区别：重载后缀时必须多一个虚拟参数：`int`，因此从形式上看像是一个二元运算符重载。

5.重载类型转换符"long"

类型转换符必须作为成员函数重载。在重载类型转换符时，由于运算符本身已经表示出返回值类型，因此不需要返回值类型的声明。一个分数可以看成是由一个整数部分和一个纯分数部分组成的，为了取得一个分数的整数部分，可为fraction重载类型转换符long。为此可在fraction.h的类声明中增加：`opertator`

`long()const { return num/den ; }` 6.重载赋值运算符"="赋值运算符只能作为成员函数重载。常见的真正需要重载赋值运算符的情况是：类中包含指向动态空间的指针赋值运算符=的重载应注意以下几点： 返回值声明为引用，而函数体中总是用语句`return *this ;`返回； 如果参数被声明为指向同类对象的引用或指针，应判别所指向对象的是否与被赋值对象为同一对象，如果是，立即返回，不做任何赋值处理； 如果被赋值对象占用了动态空间或其他资源，应首先释放这些资源，以便接收新的资源； 如果参数被声明为指针或引用，通常应加上`const`修饰； 如果参数被声明为指针，应判别是否为空，以便做出特殊处理； 一个类如果需要重载运算符=，通常也就需要定义自己特有的拷贝构造函数，反之亦然。

100Test 下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com