

码农时代

直挂云帆济沧海

[目录视图](#)[摘要视图](#)[RSS](#) [订阅](#)

个人资料



wanjun8659

访问：92664次

积分：1614

等级：

[博客Markdown编辑器上线啦](#) [那些年我们追过的Wrox精品红皮计算机图书](#) [PMBOK第五版精讲视频教程](#) [火星敏捷开发1001问](#)

C++转换构造函数和隐式转换函数

分类：[C/C++](#)

2013-11-18 14:30

1245人阅读

[评论\(0\)](#)

[收藏](#)

[举报](#)

[转换构造函数](#)

[隐式转换函数](#)

[目录\(?\)](#)

[\[+\]](#)

今天是第一次听到C++还有个转换构造函数，之前经常见到默认构造函数、拷贝构造函数、析构函数，但是从没听说过转换构造函数，隐式转换函数也是一样，C++的确是够博大精深的，再次叹服！

其实我们已经在C/C++中见到过多次标准类型数据间的转换方式了，这种形式用于在程序中将一种指定的数据转换成另一指定的类型，也即是强制转换，比如：`int a = int(1.23)`，其作用是将1.23转换为整形1。然而对于用户自定义的类类型，编译系统并不知道如何进行转换，所以需要定义专门的函数来告诉编译系统该如何转换，这就是转换构造函数和类型转换函数！

排名： 第11297名

原创： 56篇 转载： 2篇

译文： 0篇 评论： 74条

文章搜索

文章分类

Java (1)

Interview (0)

C/C++ (39)

Hodgepodge (1)

Algorithm (18)

ACM (13)

Android (2)

C# (15)

IT风云 (0)

多线程 (5)

设计模式 (1)

文章存档

2014年03月 (1)

2013年11月 (18)

2013年01月 (1)

2012年10月 (1)

2012年09月 (1)

一、转换构造函数

转换构造函数(conversion constructor function) 的作用是将一个其他类型的数据转换成一个类的对象。

当一个构造函数只有一个参数，而且该参数又不是本类的const引用时，这种构造函数称为转换构造函数。

转换构造函数是对构造函数的重载。

例如：

```
[cpp] view plain copy print ?
01.     Complex(double r)
02.     {
03.         real=r;
04.         imag=0;
05.     }
```

其作用是将double型的参数r转换成Complex类的对象,将r作为复数的实部,虚部为0。用户可以根据需要定义转换构造函数,在函数体中告诉编译系统怎样去进行转换。

那么如何使用转换构造函数进行类型转换呢？我们看如下的例子：

```
[cpp] view plain copy print ?
01.     // TypeSwitch.cpp : 定义控制台应用程序的入口点。
02.     //
03.
04.     #include "stdafx.h"
05.     #include <iostream>
06.
07.     using namespace std;
08.
09.     class Complex
10.     {
```

[展开](#)

阅读排行

[\[C#学习笔记之多线程2\]第 \(6419\)](#)[Android如何打开/关闭系 \(5722\)](#)[\[C#学习笔记之多线程1\]第 \(4845\)](#)[C#将Excel数据表导入SC \(4441\)](#)[C++快速排序之sort\(\) \(4193\)](#)[C#解决MDI窗体闪屏的方 \(4174\)](#)[\[C#学习笔记之异步编程\] \(3984\)](#)[浅谈C#插件式程序的开发 \(3637\)](#)[C#内存映射文件学习 \(2634\)](#)[c#插件式程序开发\(二\):如 \(2262\)](#)

评论排行

[C#解决MDI窗体闪屏的方 \(11\)](#)[《Effective C++》item9: \(8\)](#)[Android如何打开/关闭系 \(8\)](#)[C++的"接口与实现分离" \(7\)](#)[\[C#学习笔记之异步编程\] \(4\)](#)[\[C#学习笔记之多线程2\]第 \(4\)](#)[C++"前置声明"那些事儿 \(3\)](#)[C#将Excel数据表导入SC \(3\)](#)[C++快速排序之sort\(\) \(2\)](#)[一个编译错误引发的对de \(2\)](#)

```
11. public:
12.     Complex():real(0),imag(0){};
13.     Complex(double r, double i):real(r),imag(i){};
14.     Complex(double r):real(r),imag(0){}; // 定义转换构造函数
15.
16.     void Print(){
17.         cout<<"real = " << real <<" image = " <<imag<<endl;
18.     }
19.     Complex& operator+(Complex c){
20.         return Complex(this->real + c.real, this->imag + c.imag);
21.     }
22. private:
23.     double real;
24.     double imag;
25. };
26.
27. int main(int argc, _TCHAR* argv[])
28. {
29.     Complex c;
30.     c = 1.2; // 调用转换构造函数将1.2转换为Complex类型
31.     c.Print();
32.     Complex c1(2.9, 4.2);
33.     Complex c2 = c1 + 3.1; // 调用转换构造函数将3.1转换为Complex类型
34.     c2.Print();
35.     return 0;
36. }
```

不仅可以将一个标准类型数据转换成类对象,也可以将另一个类的对象转换成转换构造函数所在的类对象。如可以将一个学生类对象转换为教师类对象,可以在Teacher类中写出下面的转换构造函数:

```
[cpp] view plain copy print ?
01. Teacher(Student& s)
02. {
03.     num=s.num;
04.     strcpy(name,s.name);
```

推荐文章

- * CSS变量试玩儿
- * 【Android开发经验】兼容不同的屏幕大小
- * Cocoa Core Competencies_1_Accessibility
- * QtAndroid详解(3): startActivity实战Android拍照功能
- * PHPer都应该关注的服务端性能问题-听云Server试用笔记

最新评论

C#解决MDI窗体闪屏的方法

张赠华: 这个问题真困扰, 不单是MDI, 普通窗口也这样

C#解决MDI窗体闪屏的方法

张赠华: WIN7可以, 效果相当于全部绘制好后一起呈现, 但在XP下绘制出问题了

[ACM]编程珠玑

zsy5726: 你的错了, 给一个我的用例: 11 bbbbrbrbrbr

C#解决MDI窗体闪屏的方法

forcyever: 不管是否会出现其他问题, 至少这个解决了窗体闪烁问题, 比很多各种转帖自己也不知道是否可行的人靠谱多了, ...

Android如何打开/关闭系统解锁屏

Joe_n: 博主, 我也是这样做的, 问题按关机键的时候小米锁屏先锁住了, 这样以来造成了二次锁屏(自己的锁屏跟小...

C#解决MDI窗体闪屏的方法

T-TOP-HELL: 这个代码会让窗体

```
05.         sex=s.sex;
06.     }
```

使用方法同上！

注意：

1.用转换构造函数可以将一个指定类型的数据转换为类的对象。但是不能反过来将一个类的对象转换为一个其他类型的数据(例如将一个Complex类对象转换成double类型数据)。

2.如果不想让转换构造函数生效，也就是拒绝其它类型通过转换构造函数转换为本类型，可以在转换构造函数前面加上explicit！例如：

```
[cpp] view plain copy print ?
01. // TypeSwitch.cpp : 定义控制台应用程序的入口点。
02. //
03.
04. #include "stdafx.h"
05. #include <iostream>
06.
07. using namespace std;
08.
09. class Complex
10. {
11. public:
12.     Complex():real(0),imag(0){};
13.     Complex(double r, double i):real(r),imag(i){};
14.     explicit Complex(double r):real(r),imag(0){}; // explicit禁止构造函数的转换功能
15.
16.     void Print(){
17.         cout<<"real = " << real <<" image = "<<imag<<endl;
18.     }
19. private:
20.     double real;
21.     double imag;
22. };
23.
```

所有的动作都变得很慢。很不流畅

C#解决MDI窗体闪屏的方法

随风随恨: 问题解决啦!!! 太牛X了, 感谢感谢!!

C#将Excel数据表导入SQL数据库

宇文小叶: 好像不太懂额。。。

C#将Excel数据表导入SQL数据库

comcyd: 屁爆了

C#解决MDI窗体闪屏的方法

dg乱舞春秋: win7 好像不行。。。。求解。。。

```
24. int main(int argc, _TCHAR* argv[])
25. {
26.     Complex c1(1.2, 2.3), c2;
27.     double d;
28.     d = c1 + 1.1; // 调用类型转换函数将c1转换为double, 编译出错!
29.     cout<<d<<endl;
30.
31.     return 0;
32. }
```

二、类型转换函数

用转换构造函数可以将一个指定类型的数据转换为类的对象。但是不能反过来将一个类的对象转换为一个其他类型的数据(例如将一个Complex类对象转换成double类型数据)。而类型转换函数就是专门用来解决这个问题的!

类型转换函数的作用是将一个类的对象转换成另一类型的数据。

如果已声明了一个Complex类,可以在Complex类中这样定义类型转换函数:

```
[cpp] view plain copy print ?
01. operator double( )
02. {
03.     return real;
04. }
```

类型转换函数的一般形式为:

operator 类型名()

{

实现转换的语句

}

注意事项:

- 1.在函数名前面不能指定函数类型,函数没有参数。
- 2.其返回值的类型是由函数名中指定的类型名来确定的。
- 3.类型转换函数只能作为成员函数,因为转换的主体是本类的对象,不能作为友元函数或普通函数。
- 4.从函数形式可以看到,它与运算符重载函数相似,都是用关键字operator开头,只是被重载的是类型名。

double类型经过重载后,除了原有的含义外,还获得新的含义(将一个Complex类对象转换为double类型数据,并指定了转换方法)。这样,编译系统不仅能识别原有的double型数据,而且还会把Complex类对象作为double型数据处理。

[\[cpp\]](#) [view](#) [plain](#) [copy](#) [print](#) ?

```
01. // TypeSwitch.cpp : 定义控制台应用程序的入口点。
02. //
03.
04. #include "stdafx.h"
05. #include <iostream>
06.
07. using namespace std;
08.
09. class Complex
10. {
11. public:
12.     Complex():real(0),imag(0){};
13.     Complex(double r, double i):real(r),imag(i){};
14.     Complex(double r):real(r),imag(0){}; // 定义转换构造函数
15.
16.     void Print(){
17.         cout<<"real = " << real <<" image = "<<imag<<endl;
18.     }
19.     operator double(){ // 定义类型转换函数
20.         return real;
21.     }
```

```
22.     private:
23.         double real;
24.         double imag;
25.     };
26.
27.     int main(int argc, _TCHAR* argv[])
28.     {
29.         Complex c1(1.2, 2.3);
30.         double d;
31.         d = c1 + 1.1; // 调用类型转换函数将c1转换为double
32.         cout<<d<<endl;
33.
34.         return 0;
35.     }
```

本例中，对于d = c1 + 1.1；先调用类型转换函数将c1转为double类型，然后在与1.1相加！

那么程序中的Complex类对具有双重身份,既是Complex类对象,又可作为double类型数据。Complex类对象只有在需要时才进行转换,要根据表达式的上下文来决定。转换构造函数和类型转换运算符有一个共同的功能: 当需要的时候,编译系统会自动调用这些函数,建立一个无名的临时对象(或临时变量)。

上一篇 [《Effective C++》item11：在operator=中处理“自我赋值”](#)

下一篇 [《Effective C++》item25：考虑写出一个不抛异常的swap函数](#)

顶 踩
0 0

主题推荐

[c++](#)[应用程序](#)[控制台](#)[标准](#)[function](#)

猜你在找

[LuaBind --最强大的Lua C++ Bind](#)[LuaBind --最强大的Lua C++ Bind](#)[LuaBind --最强大的Lua C++ Bind](#)[LuaBind --最强大的Lua C++ Bind](#)[C++术语对照](#)[C++术语对照](#)[C++链接器工具错误LNK2001 LNK2019](#)[C++技术总结](#)[C++术语对照](#)[java基础教程-网络应用](#)

准备好了么？ **跳** 吧！

更多职位尽在 **CSDN JOB**

[软件开发工程师（C++）](#)[我要跳槽](#)[C++服务器](#)[我要跳槽](#)[苏州敏行医学信息技术有限公司](#)[| 6-10K/月](#)[北京乐动卓越科技有限公司](#)[| 15-20K/月](#)

C++软件开发工程师

我要跳槽

软件工程师 (C++/C#)

我要跳槽

天津市努思企业服务有限公司

| 7-10K/月

武汉海翼科技有限公司

| 4-6K/月

查看评论

暂无评论

您还没有登录,请[登录](#)或[注册](#)

* 以上用户言论只代表其个人观点,不代表CSDN网站的观点或立场

核心技术类目

全部主题 Hadoop AWS 移动游戏 Java Android iOS Swift 智能硬件 Docker
OpenStack VPN Spark ERP IE10 Eclipse CRM JavaScript 数据库 Ubuntu NFC
WAP jQuery BI HTML5 Spring Apache .NET API HTML SDK IIS Fedora XML
LBS Unity Splashtop UML components Windows Mobile Rails QEMU KDE Cassandra
CloudStack FTC coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App
SpringSide Maemo Compuware 大数据 aptech Perl Tornado Ruby Hibernate ThinkPHP
HBase Pure Solr Angular Cloud Foundry Redis Scala Django Bootstrap

公司简介 | 招贤纳士 | 广告服务 | 银行汇款帐号 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服

杂志客服

微博客服

webmaster@csdn.net

400-600-2320 | 北京创新乐知信息技术有限公司 版权所有 | 江苏乐知网络技术有限公司 提供商务支持

