

# 飞空静渡

[目录视图](#)[摘要视图](#)[RSS](#) [订阅](#)

## 个人资料



fjb2080

访问：512884次

积分：7078

等级：

排名：第1042名

[HTML 5全掌控](#) [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](#) [CSDN JOB带你坐飞机回家过年](#)

## c++之类的前置声明

2010-04-27 12:36

2473人阅读

[评论\(2\)](#)[收藏](#)[举报](#)

c++

class

c

原创文章，转载请注明出处，谢谢！

作者：清林，博客名：飞空静渡

刚开始学习c++的人都会遇到这样的问题：

定义一个类 class A，这个类里面使用了类B的对象b，然后定义了一个类B，里面也包含了一个类A的对象a，就成了这样：

原创： 161篇 转载： 150篇

译文： 0篇 评论： 154条

## 文章搜索

## 文章分类

BIOS (3)

boost (2)

C/C++ (79)

gcc (1)

linux (66)

MeeGo (4)

QT (48)

QT浏览器 (2)

WINDOWS (24)

个人作品 (18)

大话人生 (8)

实战DeviceIoControl (7)

数据恢复 (20)

计算机技术 (20)

设计模式 (3)

汇编 (1)

loki (1)

Linux Shell (9)

vim (0)

C++11 (31)

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

```
01. //a.h
02. #include "b.h"
03. class A
04. {
05. ....
06. private:
07.     B b;
08. };
09. //b.h
10. #include "a.h"
11. class B
12. {
13. ....
14. private:
15.     A a;
16. };
```

一编译，就出现了一个互包含的问题了，这时就有人跳出来说，这个问题的解决办法可以这样，在a.h文件中声明类B，然后使用B的指针。

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

```
01. //a.h
02. // #include "b.h"
03. class B;
04. class A
05. {
06. ....
07. private:
08.     B b;
09. };
10. //b.h
11. #include "a.h"
```

## 文章存档

2014年06月 (1)

2014年05月 (2)

2014年04月 (1)

2013年11月 (6)

2013年10月 (1)

展开

## 阅读排行

C++11 FAQ中文版: std: (14817)

手把手教你用WinHex在 (10746)

4KB大扇区硬盘的性能分 (9647)

QuickLZ —— 一个号称世 (9598)

QListWidget和QListWidg (8791)

关于编译glibc-2.9的The (8500)

QT中的pro文件的编写 (8378)

QT浏览器学习之一: QW (8345)

QTableWidget的简单使 (7741)

深入浅出linux内核源代码 (7227)

## 评论排行

NTFSRecover 1.3 全新改 (12)

c++之指针作为函数参数 (11)

深入浅出linux内核源代码 (9)

c语言中的函数间的跳转 (4)

```
12.  class B
13.  {
14.      ....
15.  private:
16.      A a;
17.  };
```

然后, 问题就解决了。

但是, 有人知道问题是为什么就被解决的吗, 也就是说, 加了个前置声明为什么就解决了这样的问题。下面, 让我来探讨一下这个前置声明。

类的前置声明是有许多的好处的。

我们使用前置声明的一个好处是, 从上面看到, 当我们在类A使用类B的前置声明时, 我们修改类B时, 只需要重新编译类B, 而不需要重新编译a.h的 (当然, 在真正使用类B时, 必须包含b.h)。

另外一个好处是减小类A的大小, 上面的代码没有体现, 那么我们来看下:

```
[cpp] view plain copy print ?
01.  //a.h
02.  class B;
03.  class A
04.  {
05.      ....
06.  private:
07.      B *b;
08.      ....
09.  };
10.  //b.h
11.  class B
12.  {
13.      ....
```

- 简易分析myicq的内存池 (4)
- QT浏览器学习之一: QW (4)
- 不能为虚拟电脑 xxx 打开 (3)
- QTreeWidget的简单使用 (3)
- ext4分区中恢复被删除的 (3)
- 深入浅出linux内核源代码 (3)

### 推荐文章

- \* 浅析总结 Java 内部类的一些使用与梳理
- \* Qt for iOS, Qt 与 Objective C 混合编程
- \* 教你写Android ImageLoader框架之基本架构
- \* 三大运营商的游戏“刷金”漏洞解决方案
- \* 百度地图开发 (二) 之添加覆盖物
- ⬇ 地理编码和反地理编码

### 最新评论

c++之指针作为函数参数传递的问题  
tengfei461807914: 讲的太好了  
~~

c++之指针作为函数参数传递的问题  
edcapding: @sunner5222:看的时候就比较疑惑这里, 后来运行楼主原错误代码int \*p=x; \*x = \*...

通过给程序里嵌入manifest, 使得  
qiqi777a: 可否通过此方式, 让程序不弹出uac窗口?

QListWidget和QListWidgetItem

```
14.     private:
15.         int a;
16.         int b;
17.         int c;
18.     };
```

我们看上面的代码, 类B的大小是12 (在32位机子上)。

如果我们在类A中包含的是B的对象, 那么类A的大小就是12 (假设没有其它成员变量和虚函数)。如果包含的是类B的指针\*b变量, 那么类A的大小就是4, 所以这样是可以减少类A的大小的, 特别是对于在STL的容器里包含的是类的对象而不是指针的时候, 这个就特别有用了。

在前置声明时, 我们只能使用的就是类的指针和引用 (因为引用的实现的)。

那么, 我问你一个问题, 为什么我们前置声明时, 只能使用类型 呢?

如果你回答到: 那是因为指针是固定大小, 并且可以表示任意的类型, 那么可以给你80分了。为什么只有80分, 因为还没有完全回答到。

想要更详细的答案, 我们看下下面这个类:

```
[cpp] view plain copy print ?
01.     class A
02.     {
03.     public:
04.         A(int a):_a(a),_b(_a){} // _b is new add
05.
06.         int get_a() const {return _a;}
07.         int get_b() const {return _b;} // new add
```

\_hello\_word: 怎样在 QListWidgetItem 中的已经存在的 QIcon 上画图？比如说在原有的图片上加个对号，怎...

c++之指针作为函数参数传递的问  
ACder: 太感谢了，我找的很多资料都没有解决这个问题，今天终于明白了，楼主给力！！！！

QT内置的ICON资源  
cliffordl: 怎么使用啊，获取 QIcon？

c++之指针作为函数参数传递的问  
冬天-来了: 写的太好了了，通俗易懂

QT中的元对象系统（一）：QVariant  
blueice12: Q\_DECLARE\_METATYPE

程序猿眼中的分区与国家的关系  
galileo\_X: 楼主倒是说得很好，不过还是没能解决我的疑惑，还是赞一个啦

QListWidget和QListWidgetItem  
lianghongge: 很有用，谢谢

好友

亮哥

```
08.     private:
09.         int _b; // new add
10.         int _a;
11.     };
```

我们看下上面定义的这个类A，其中\_b变量和get\_b()函数是新增加进这个类的。

那么我问你，在增加进\_b变量和get\_b()成员函数后这个类发生了什么改变，思考一下再回答。

好了，我们来列举这些改变：

第一个改变当然是增加了\_b变量和get\_b()成员函数；

第二个改变是这个类的大小改变了，原来是4，现在是8。

第三个改变是成员\_a的偏移地址改变了，原来相对于类的偏移是0，现在是4了。

上面的改变都是我们显式的、看得到的改变。还有一个隐藏的改变，想想是什么。。。

这个隐藏的改变是类A的默认构造函数和默认拷贝构造函数发生了改变。

由上面的改变可以看到，任何调用类A的成员变量或成员函数的行为都需要改变，因此，我们的a.h需要重新编译。

如果我们的b.h是这样的：

```
[cpp] view plain copy print ?
01.     //b.h
02.     #include "a.h"
03.     class B
```

```
04. {  
05. ...  
06. private:  
07.     A a;  
08. };
```

那么我们的b.h也需要重新编译。

如果是这样的：

```
[cpp] view plain copy print ?  
01. //b.h  
02. class A;  
03. class B  
04. {  
05. ...  
06. private:  
07.     A *a;  
08. };
```

那么我们的b.h就不需要重新编译。

像我们这样前置声明类A：

```
class A;
```

是一种不完整的声明，只要类B中没有执行需要了解类A的大小或者成员的操作，则这样的不完整声明允许声明指向A的指针和引用。

而在前一个代码中的语句

```
A a;
```

是需要了解A的大小的，不然是不可能知道如果给类B分配内存大小的，因此不完整的前置声明就不行，必须要包含a.h来获得类A的大小，同时也要重新编译类B。

再回到前面的问题，使用前置声明只允许的声明是指针或引用的一个原因是只要这个声明没有执行需要了解类A的大小或者成员的操作就可以了，所以声明成指针或引用是没有执行需要了解类A的大小或者成员的操作的。

上一篇 [c++语法之switch语句](#)

下一篇 [通过\\$Bitmap元文件计算NTFS分区的使用空间大小](#)

## 主题推荐

[c++](#)[类](#)[32位](#)[对象](#)[博客](#)

## 猜你在找

[SQL Server随机抽取一行针对有索引或主键的表](#)[简易却不失华丽 进销存管理 \] BCB+ACCESS](#)[一步步进行 LUA的OOD封装 一](#)[创建有个性的对话框之MFC篇一](#)[Linux网络编程之使用TCP传输文件](#)[课程设计报告](#)[POCOFoundation 内存管理一 AutoPtr](#)[QT 设置程序界面风格](#)[C++ 日志库log4cplus使用手册](#)[Delphi 使用帮助提示Hint制作简单即时的系统帮助](#)

## 准备好了么？跳吧

[更多职位尽在 CSDN JOB](#)[c++开发工程师](#)[我要跳槽](#)[C++服务器](#)[我要跳槽](#)[京品高科信息科技（北京）有限公司](#)[| 7-15K/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](#)



### 查看评论

2楼 [jiangwlee](#) 2012-03-28 17:19发表



第二段代码是编译不过的，后面说了，前置声明之后，只能使用指针或者引用。

1楼 [flyingfishouc](#) 2011-03-08 23:28发表



在第二段代码中，请问8. B b;这一句，是如何知道B的大小的呢？

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



2015年2月5日

c++之类的前置声明 - 飞空静渡 - 博客频道 - CSDN.NET

[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](mailto: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

京 ICP 证 070598 号 | Copyright © 1999-2014, CSDN.NET, All Rights Reserved

