

Qt下监测内存泄漏的方法

作者：有故事的人 来源：范文网 www.wtabcd.cn/fanwen/

本文原地址：<https://www.wtabcd.cn/fanwen/zuowen/48edb39ba01669cec47544ad23e2bc55.html>

范文网，为你加油喝彩！

在写qt应用程序时，由于是采用c++语言，经常会碰到一个令人棘手的问题，那就是内存泄漏，虽然面c++为了防止内存泄漏，发布了智能指针以用来避免内存泄漏，但是并不能完全避免。而且智能指针使用不当，同样会造成非常严重的问题。这里智能指针就不在赘述，有机会我再写一篇智能指针的博客。

为了预防内存泄漏问题，我们手动写一个宏，用来监测该类是否析构，若未析构，则将该类打印出来，这样可以更好的定位问题。代码如下：

定义一个帮助类，用来计数，当注册的类声明时，计数+1，当类析构时，计数-1，若在该类析构时，计数大于0，则该注册的类未析构，则打印出来，提示开发者出现内存泄漏问题，告诉是哪个类未析构

```
class countcheckerhelper{public: countcheckerhelper(const char *classname, const char
*filename) : m_classname(classname) , m_filename(filename)
, m_refcount(0) { } ~countcheckerhelper() {
if (m_refcount > 0) { qDebug()
<< m_filename << m_classname << "is not delete..."; } }
void operator ++() { ++m_refcount; } void ope
rator --() { --m_refcount; }private: cons矢量叉乘t ch
ar *m_classname; const char 古剑情韵*m_filename; unsigned int m_refcount
};#define capture_memory_leak(classname) \class countchecker
{
\public:
\ coun谁的青春不腐朽tchecker() \ {
\ +
+counter(); \ }
\ ~countchecker()
\ {
\ --counter(); \
} \private:
\ countcheckerhelper
&counter() \ {
\ static countcheckerhelper instance(#classname, __file__);
\ return instance; \ }
```

```
\} countchecker;  
\
```

利用static局部变量的生存周继发性主动转运期，当程序退出时，静态局部变量析构，此时countcheckerhelper析构，若此时注册的类的计数大于0，则证明该注册的类内存泄漏，将该宏写入到你的类的如下位置，其实就是声明一个内部类，并创建这个内部类的对象。

```
class object{ capture_memory_leak(object)public: object() { }}; i  
nt main(){ objec今天明天t *object = new object(); return 0;}
```

结果：打印出内存泄漏的文件，以及泄漏的类名object

到此这篇关于qt下监测内存泄漏的方法的文章就介绍到这了,更多相关qt 监测内存泄漏内容请搜索www.887551.com以前的文章或继续浏览下面的相关文章希望大家以后多多支持www.887551.com！

更多 作文 请访问 https://www.wtabcd.cn/fanwen/list/92_0.html

文章生成doc功能，由[范文网](#)开发