

清风小阁

于小阁，守望星空。

个人资料



huang_xw

访问： 984379次

积分： 13344

等级： BLOG > 7

排名： 第310名

原创： 348篇 转载： 183篇

译文： 1篇 评论： 167条

文章搜索

文章分类

[路过人间] (39)

[WEB] (17)

[JavaScript] (10)

[CSS] (15)

[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](#) [火星人敏捷开发1001问](#)

【C++11】新特性——auto的使用

分类： [C++]

2013-04-06 06:39

11672人阅读

评论(6)

收藏

举报

目录(?)

[+]

C++11中引入的auto主要有两种用途：自动类型推断和返回值占位。auto在C++98中的标识临时变量的语义，由于使用极少且多余，在C++11中已被删除。前后两个标准的auto，完全是两个概念。

1. 自动类型推断

auto自动类型推断，用于从初始化表达式中推断出变量的数据类型。通过auto的自动类型推断，可以大大简化我们的编程工作。下面是一些使用auto的例子。

```
[cpp] view plain copy print ?
01. #include <vector>
02. #include <map>
03.
04. using namespace std;
05.
06. int main(int argc, char *argv[], char *env[])
07. {
08.     // auto a;           // 错误，没有初始化表达式，无法推断出a的类型
09.     // auto int a = 10;   // 错误，auto临时变量的语义在C++11中已不存在，这是旧标准的用法。
10.
11.     // 1. 自动帮助推导类型
12.     auto a = 10;
13.     auto c = 'A';
14.     auto s("hello");
15.
16.     // 2. 类型冗长
17.     map<int, map<int,int> > map_;
18.     map<int, map<int,int>>::const_iterator itr1 = map_.begin();
19.     const auto itr2 = map_.begin();
20.     auto ptr = []()
```

[delphi] (14)
[JAVA] (27)
[JAVA]--[网络编程] (13)
[JAVA]--[并发编程] (21)
[JAVA]--[源码赏析] (0)
[JAVA]--[设计模式] (3)
[C++] (51)
[C++]--[STL] (36)
[C++]--[Boost] (50)
[C++]--[MFC] (4)
[C++]--[并发编程] (17)
[C++]--[Error] (14)
[C++]--[Poco] (16)
[C++]--[Dom] (3)
[Linux]--[系统管理] (40)
[Linux]--[命令] (8)
[数据库设计] (9)
[MySQL] (9)
[oracle]--[管理] (58)
[oracle]--[PL/SQL] (14)
[oracle]--[存储结构] (19)
[oracle]--[调优] (8)
[oracle]--[备份与恢复] (9)
[oracle]--[用户与安全] (6)
[oracle]--[Error] (24)
[oracle]--[实验] (14)
[C#与dot net] (15)
[其他] (6)

文章存档

2013年12月 (4)

2013年10月 (16)

2013年09月 (4)

2013年08月 (5)

2013年07月 (5)

展开

阅读排行
搜索百度云资源的一点小 (47860)
Java多线程之Lock的使用 (19873)
国内速度比较快的NTP S (16954)
oracle impdp的table_ex (13687)
【C++11】新特性——at (11626)
ping: unknown host 解决 (9241)
java读取某目录下所有文 (8902)

```
21.     {
22.         std::cout << "hello world" << std::endl;
23.     };
24.
25.     return 0;
26. };
27.
28. // 3. 使用模板技术时，如果某个变量的类型依赖于模板参数，
29. // 不使用auto将很难确定变量的类型（使用auto后，将由编译器自动进行确定）。
30. template <class T, class U>
31. void Multiply(T t, U u)
32. {
33.     auto v = t * u;
34. }
```

2. 返回值占位

```
[cpp] view plain copy print ?
01. template <typename T1, typename T2>
02. auto compose(T1 t1, T2 t2) -> decltype(t1 + t2)
03. {
04.     return t1+t2;
05. }
06. auto v = compose(2, 3.14); // v's type is double
```

3.使用注意事项

①我们可以使用valatile，pointer（*），reference（&），rvalue reference（&&）来修饰auto

```
[cpp] view plain copy print ?
01. auto k = 5;
02. auto* pK = new auto(k);
03. auto** ppK = new auto(&k);
04. const auto n = 6;
```

②用auto声明的变量必须初始化

```
[cpp] view plain copy print ?
01. auto m; // m should be intialized
```

③auto不能与其他类型组合连用

```
[cpp] view plain copy print ?
01. auto int p; // 这是旧auto的做法。
```

④函数和模板参数不能被声明为auto

```
[cpp] view plain copy print ?
01. void MyFunction(auto parameter){} // no auto as method argument
02.
03. template<auto T> // utter nonsense - not allowed
04. void Fun(T t){}
```

随书光盘免费下载的好地	(8901)
IT牛人博客	(8475)
【Boost】boost::noncop	(8174)

评论排行	
Microsoft Visual Studio 2	(8)
VS2010检测到"_ITERAT	(8)
SOCI的编译注意事项（	(7)
OCX调用页面上JavaScr	(6)
【C++11】新特性——at	(6)
IT牛人博客	(6)
oracle中PCTFREE, PCT	(6)
Oracle 用户、角色、权限	(5)
Eclipse下安装Aptana St	(5)
【Boost】boost库asio详	(5)

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	上 地理信息和后地理信息

最新评论	
【Boost】boost库中thread多线程	继续微笑lsj: 有个小问题请教一下，boost提供了这个thread库，在linux下选择posix线程库和这个库有...
ORA-28002: the password will e	stubborn超: 大哥，你的alter user system identified by root; 这句话有什么作...
IT牛人博客	tanghai_lin: 顶一个
【C++11】新特性——auto的使	skyztttt: 讲得很透彻，支持~~
TVideoCapture类的源码(Delphi	houjbf: 偶尔会弹出“选择视频源”对话框，用起来不方便，这个怎么解决？
【C++11】新特性——auto的使	wangzhe03091252: auto ptr = [] () { std::cout << "hell...

⑤定义在堆上的变量，使用了**auto**的表达式必须被初始化

	[cpp] view plain copy print ?
01.	int * p = new auto(0); //fine
02.	int * pp = new auto(); // should be initialized
03.	
04.	auto x = new auto(); // Hmmm ... no intializer
05.	
06.	auto* y = new auto(9); // Fine. Here y is a int*
07.	auto z = new auto(9); //Fine. Here z is a int* (It is not just an int)

⑥以为**auto**是一个占位符，并不是一个他自己的类型，因此不能用于类型转换或其他一些操作，如**sizeof**和**typeid**

	[cpp] view plain copy print ?
01.	int value = 123;
02.	auto x2 = (auto)value; // no casting using auto
03.	
04.	auto x3 = static_cast <auto>(value); // same as above

⑦定义在一个**auto**序列的变量必须始终推导成同一类型

	[cpp] view plain copy print ?
01.	auto x1 = 5, x2 = 5.0, x3='r'; // This is too much....we cannot combine like this

⑧**auto**不能自动推导成CV-qualifiers（constant & volatile qualifiers），除非被声明为引用类型

	[cpp] view plain copy print ?
01.	const int i = 99;
02.	auto j = i; // j is int, rather than const int
03.	j = 100 // Fine. As j is not constant
04.	
05.	// Now let us try to have reference
06.	auto& k = i; // Now k is const int&
07.	k = 100; // Error. k is constant
08.	
09.	// Similarly with volatile qualifer

⑨**auto**会退化成指向数组的指针，除非被声明为引用

	[cpp] view plain copy print ?
01.	int a[9];
02.	auto j = a;
03.	cout<< typeid (j).name()<<endl; // This will print int*
04.	
05.	auto& k = a;
06.	cout<< typeid (k).name()<<endl; // This will print int [9]

vs2005添加附加库目录或者库文
追寻灯火阑珊: 请教一个问题 我
按上述方法已经正确设置了 还是
不可以, 说 找不到sqlite3.lib库
文件, ...

【Boost】利用typeid来获取变量
郝立杰hlj: int_fast_t是什么作用
啊? 还有就是这段程序需要加什
么头文件呢?

Microsoft Visual Studio 2010 遇
快乐起航2020: 我的也是

【C++11】新特性——auto的使
nscboy: @Kurodo:不是托管,只
是简化类型声明.自己new的对象
还得自己释放.否则会有内存泄漏
的.

上一篇 【Boost】Boost总结汇总

下一篇 【C++11】新特性——引入nullptr

顶
8

踩
0

主题推荐

c++11

namespace

编译器

标准

iterator

猜你在找

string的size和length

Boostboost库asio详解3io_service作为work pool

xcode模拟器调试无法显示调用栈的解决办法

C++学习之深入理解虚函数一虚函数表解析

自己选择的路跪着也要走完

moto & google笔试题目-STLC++面试题

vim显示行号语法高亮自动缩进的设置

抄袭事件判决书

面试经验

静态库和动态库里的全局变量或类的静态成员变量共享

查看评论

5楼 skyztttt 2014-12-24 17:02发表



讲得很透彻，支持~~

4楼 wangzhe03091252 2014-12-18 12:02发表



auto ptr = []()
{
std::cout << "hello world" << std::endl;
};
楼主，你好。ptr是个函数指针吧。后面的[]()是什么意思。

3楼 Tangbzh 2014-07-07 23:10发表



不错

2楼 Kurodo 2014-05-12 22:09发表



auto 是不是托管，自动释放

Re: nscboy 2014-11-20 18:34发表



回复Kurodo： 不是托管,只是简化类型声明.自己new的对象还得自己释放.否则会有内存泄漏的.

1楼 maggot611 2014-04-01 20:02发表



写得简单明了，赞一个

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