



博客

下载

学习

社区

C知道

GitCode

InsCode

会议

堆栈

搜索



会员中心

消息

历史

创作中心

发布



Lu_gl

关注

11



59

1

C++ 中类的实例化的方法

原创 Lu_gl 于 2020-03-26 12:49:12 发布 阅读量1.8w 收藏 59 点赞数 11

版权

文章标签: [指针](#) [堆栈](#) [c++](#) [c语言](#)



GitCode 开源社区 文章已被社区收录

加入社区

C++ 中类的实例化的方法

类的实例化按照在分配内存上面的不同大体分为在：栈中的分配内存和堆中的内存的分配，于是分为下面三种方式。

```
1 class CEmployee
2 {
3 private:
4     int age;
5     int id;
6 public:
7     CEmployee()
8     {
9         cout << "这是构造函数" << endl;
10    }
11    CEmployee(int id)
12    {
13        this->id = id;
14    }
15    ~CEmployee()
16    {
17        cout << "这是析构函数" << endl;
18    }
19    void OutputName()
20    {
21        cout << this->id << endl;
22    }
23 };
```

复制

一、隐式创建



Lu_gl 关注

👍 11 🗨 0 ⭐ 59 💬 1

隐式创建方式在进程虚拟地址空间中的栈中分配内存，它的分配和释放由系统决定，**函数** 内局部变量的存储单元可以在栈上创建，函数执行完毕，系统会自动释放这些存储单元。

```
1 | int main() {
2 |     CEmployee cEmployee1;    //隐式创建并调用无参构造器
3 |     CEmployee cEmployee2(2); //隐式创建并调用有参构造器
4 |     return 0;
5 | }
```

二、显式创建

这种创建方式和第一种一样，在进程虚拟地址空间中的栈中分配内存，它的分配和释放由系统决定，函数内局部变量的存储单元可以在栈上创建，函数执行完毕，系统会自动释放这些存储单元。

```
1 | int main() {
2 |     CEmployee cEmployee1 = CEmployee;    //显式创建并调用无参构造器
3 |     CEmployee cEmployee2 = CEmployee(2); //显式创建并调用有参构造器
4 |     return 0;
5 | }
```

三、显式new创建

这种方式使用了new关键字，在堆中分配了内存，堆上的内存分配，亦称动态内存分配。程序在运行的期间用malloc申请的内存，这部分内存由程序员自己负责管理，其生存期由开发者决定:在何时分配，分配多少，并在何时用free来释放该内存。

```
1 | int main() {
2 |     CEmployee *cEmployee1 = new CEmployee; //显式new创建并调用无参构造器
3 |     CEmployee *cEmployee2 = new CEmployee(2); //显式new创建并调用有参构造器
4 |     return 0;
5 | }
```

new创建的注意事项：

1. new创建类对象需要指针接收，一处初始化，多处使用
2. new创建类对象使用完需delete销毁
3. new创建对象直接使用堆空间，而局部不用new定义类对象则使用栈空间
4. new对象指针用途广泛，比如作为函数返回值、函数参数等
5. 频繁调用场合并不适合new，就像new申请和释放



Lu_gl

关注

👍 11



★ 59

💬 1

下面举一个简单的例子进行解释：

首先创建一个类，如下：

```
1 |     class Coordinate
2 |     {
3 |     public:
4 |         int m_iX;
5 |         int m_iY;
6 |     };
```

下面创建一个实例的对象：

(原文链接：<https://blog.csdn.net/sun980984305/article/details/52959893>)

```
1 | void main()
2 | {
3 |     Coordinate coord[3]; //栈上
4 |     coord[1].m_iX=10;
5 |     Coordinate *p=new Coordinate[3]; //堆中
6 |     p[0].m_iX=10;
7 |     //对于定义一个类指针的方法: A *p = new A; p->成员函数; p->成员变量;
8 |     // 或者 p.成员函数; p.成员变量;
9 |     p->m_iX=10;// 调用成员函数
10 |     delete []p;
11 |     p=NULL;
12 | }
```

在栈上实例化一个对象数组

Coordinate coord[3]，类名+ **数组** 的名称+数组的大小，创建完成后每一个数组元素都可以访问各自的成员，例如用coord[1]来访问数组的第二个元素，后面接点号来访问数据成员，形如coord[1].m_iX=10。这里使用完后内存是自动销毁的。

在堆中实例化一个对象数组

Coordinate *p=new Coordinate[3]，用一个指针指向一段内存，开辟内存用new运算符，这里数组大小为3，则**创建时会调用三次构造函数**。创建完成后每一个数组元素都可以访问各自的成员，例如p[0].m_iX=10是通过指针接一个索引再加点号和数据成员访问的，也可以通过p->m_iX=10来进行赋值。在这里使用完后，必须要通过delete来销毁这个对象数组，由于是一个对象数组，所以在**销毁的时候是调用了多次析构函数**的。

证明构造函数和析构函数被多次调用具体程序实例：



Lu_gl

关注

👍 11

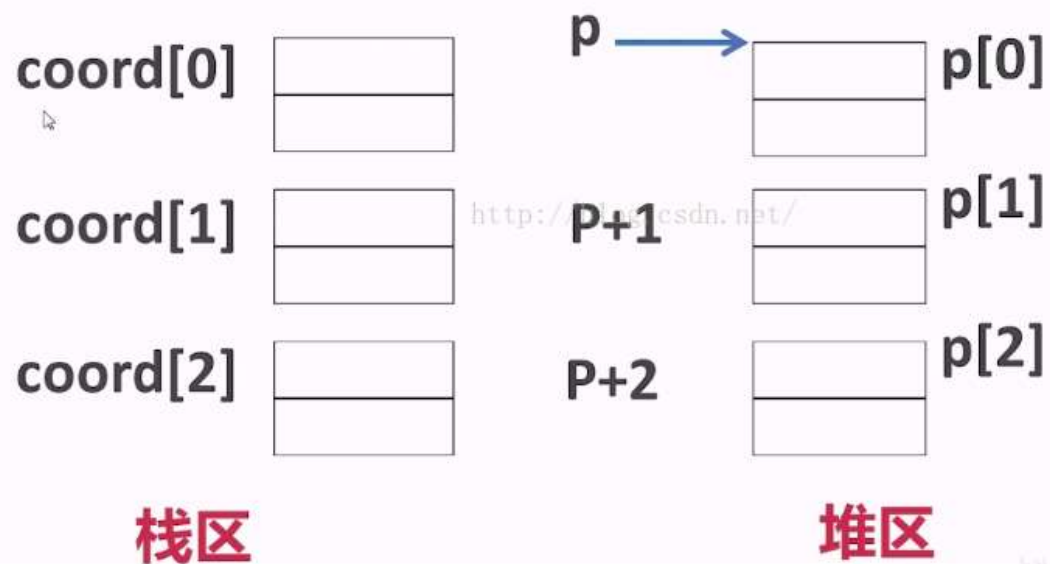


★ 59

💬 1

如图所示:

Coordinate coord[3]; Coordinate *p = new Coordinate[3];



https://blog.csdn.net/Lu_gl

```
1 Coordinate.h
2     #ifndef COORDIANTE
3     #define COORDIANTE
4     class Coordinate
5     {
6     public:
7         Coordinate();//构造函数
8         ~Coordinate();//析构函数
9     };
10    #endif
11
12    Coordinate.cpp
13    #include<iostream>
14    #include"Coordinate.h"
15    using namespace std;
16
17    Coordinate::Coordinate()
18    {
19        cout<<"Coordinate()"<<endl;
20    }
```



Lu_gl 关注

11 59 1

```

20 | }
21 |
22 | Coordinate::~Coordinate()
23 | {
24 |     cout<<"~Coordinate()"<<endl;
25 | }
26 |

```

mian.cpp

```

1 | #include<iostream>
2 | #include "Coordinate.h"
3 | using namespace std;
4 |
5 | void main()
6 | {
7 |     Coordinate coor[3]; // 隐式创建
8 |     Coordinate *p=new Coordinate[3]; // 显式new创建      创建时会调用三次构造函数
9 |     delete []p; // 销毁的时候是调用了多次析构函数
10 |    p=NULL;
11 |    system("pause");
12 | }
13 |

```

程序执行的结果:

```

Coordinate()
Coordinate()
Coordinate()
Coordinate()
Coordinate()
Coordinate()
~Coordinate()
~Coordinate()
~Coordinate()
请按任意键继续. . .

```

感谢博主: sun980984305

C++: 类的简介、如何定义并实例化一个类

使用类来定义自己的数据结构。每一个对象都有自己的属性

C++类的实例化加括号和不加括号的区别



Lu_gl

关注

qq_41657077的博客 1789

11



59

1

1 条评论



天命小脑斧 热评 最后一个案例的执行结果应该是6个析构函数吧。

写评论

C++--类的实例化_c++类的实例化

11-28

在C++ 中,即使没有显式地对类的成员变量进行“实例化”(初始化),但当定义类的对象时,依然会为对象分配内存空间以容纳类的成员变量。 在上述代码中,虽然没有显式地创建Data类的对象,但在Print函数中使...

C++内存申请、实例化的方式_c++类申请实例

11-17

对象:具体,构造函数和析构函数,用于描述对象的建立和撤销 this指针:可将类和对象有机结合 封装:把实现的细节封装起来,只暴露用户关心的部分 访问限定符:public protected private 实例化的方式:堆和栈 栈...

c++类实例化的两种方式 热门推荐

u010039733的专栏 2万+

/* HelloWorld.cpp */ Created on: 2016年7月13日 * Author: zroad */#include<iostream> #include<cstring> #include<string> using namespace std;class C { public: C(string s = "", int i = 0, d

C++：类的定义和实例化

Another_Shi的博客 1379

class为定义类的关键字，Stack为类的名字，{}中为类的主体，注意类定义结束时后面分号不能省略。类体中内容称为类的成员：类中的变量称为类的属性或成员变量；类中的函数称为类的方法或者成员函数...

C++ 模板实例化与调用_c++模块实例

11-2

1.2 类模板隐式实例化 类模板隐式实例化指的是在使用模板类时才会将模板实例化,相对于类模板显示实例化而言的。 #include<iostream>using namespace std;template<typename T>class A{T num;public:A(){n...

C++:类的定义和实例化_类的实例化

12-3

一、类的定义 1、类的定义格式 2、访问限定符 3、类域 二、实例化 1、实例化的概念 2、对象大小 一、类的定义 1、类的定义格式 class为定义类的关键字,Stack为类的名字,{}中为类的主体,注意类定义结束...

C++语法——类的实例化 最新发布

m0_73633088的博客 658

直接初始化：更高效，不涉及临时对象的拷贝。拷贝初始化：涉及临时对象的创建和拷贝，可能引入性能开销。

c++中类的实例化

Sugar_goat的博客 1158

刚刚说到C++中有struct和class两个关键字，它们的区别在于struct是默认将所有的成员公有化，class默认将所有的成员私有化，到这里，有人会问，什么是公有化？这也就说明了我们在使用c语言和c++所...

...类实例化对象的创建和释放_c++类实例化和创建对象

11-23

1. 类的实例化(创建对象)通常有两种方式 栈上实例化:通过直接声明一个对象,对象会被分配在栈上。 堆上实例化:使用 new 操作符在堆上分配内存来创建对象,返回对象的指针。(1)栈上 创建和释放:实例化为...

C++类和对象(二)——实例化、对象模型、this指针_c++实例化对象-CSDN博 ...

11-14

1. 类的实例化 用类类型创建对象的过程,称为类的实例化 类是对对象进行描述的,是一个模型一样的东西,限定了类有哪些成员,定义出一个类并没有分配实际的内存空间来存储它;比如:入学时填写的学生信息表...

C++--类的实例化

weixin_74202161的博客 1252

用类类型在屋里内存中创建对象的过程，称为类实例化出对象类是对对象进行一种抽象描述，是一个模型一样的东西，限定了类有哪些成员变量，这些成员变量只是声明，没有分配空间，用类实例化出对象...

C++类的实例化

xuwenjie123的博客 6145

在c++中定义类，实例对象通常是有两种方法：1.从栈中请求空间，实例化对象，创建一个坐标的类，代码如下： #include<iostream>;using namespace std;class Coordinate{ public: int x; int ...

C++从入门到精通——类的作用域及类的实例化_类作



Lu_gi 关注

11 59 1

通过合理地使用作用域,可以控制类中成员的访问权限,提高代码的封装性和安全性。同时,类的作用域也可以使代码更加模块化,便于理解和维护。二、类的实例化 用类的类型创建对象的过程,称为类的实例化 ...

【C++】类和对象(上):类、实例化、this指针_c++类实例化 11-27
1.实例化概念 2.对象的大小 三.this指针 一.类的定义 1.面向对象的思想 众所周知,C语言是面向过程的语言,对于每一个过程都有一个对应的函数来实现,每个函数都相对独立,但随着计算机发展,问题的场景也越...

极智Coding | C++ 类实例化的方法 极智视界 3531
本文介绍一下 C++ 中类实例化的几种方法。

C++类的实例化方法 m0_46388786的博客 4928
实例化：是指创建一个对象的过程 C++中类的实例化有两种方式——在栈中实例化和在堆中实例化 在栈中实例化为静态分配内存，不需要手动回收，超出内存系统报错。例如：Student a或者Student a[3]，...

0361-极智开发-解读C++类实例化的方法 05-06
0361_极智开发_解读C++类实例化的方法

C++中COM组件初始化方法实例分析 01-20
本文实例讲述了C++中COM组件初始化方法。分享给大家供大家参考。具体如下： 这里使用BCB 在使用TADOConnect等组件时需要进行初始化 调用接口： CoInitialize(NULL);//初始化COM套件 CoUninitia...

C++中类的实例化的两种方法 铅化内敛->比你牛逼的人比你更努力 1万+
首先创建一个类，如下： class Coordinate { public: int m_iX; int m_iY; }; 下面就是实例化一个对象。 void main() { Coordinate coord[3]; //栈上 coord[1].m_iX=10; Coordinate *p=new Coordinate[3]; //堆中 ...

c++ 类与实例化 weixin_43868552的博客 1939
c++类与实例化简单描述

C++类的实例化的两种方法 nhf_a的博客 885
/* * HelloWorld.cpp * * Created on: 2016年7月13日 * Author: zroad */ #include #include #include using namespace std; class C { public: C(string s = "", int i = 0, double d = 1.0) {

C++从入门到起飞之——类的定义&实例化 全方位剖析！ 2301_80221228的博客 2627
•class为定义类的关键字，Test为类的名字，{}中为类的主体，注意类定义结束时后面分号不能省 略。类体中内容称为类的成员：类中的变量称为类的属性或成员变量;类中的函数称为类的方法或 者成员函数...

c++ 模板的具体化与实例化 weixin_33801856的博客 1064
函数模板是C++新增的一种性质，它允许只定义一次函数的实现，即可使用不同类型的参数来调用该函数。这样做可以减小代码的书写的复杂度，同时也便于修改（注：使用模板函数并不会减少最终可执行...

C++中实例化一个类的方式 积累点滴，保持自我 2589
Class A { ... }; A a; A * a = new a(); 以上两种方式皆可实现类的实例化，有new的区别在于： 1.前者在堆栈中分配内存，后者为动态内存分配，在一般应用中是没有什么区别的，但动态内存分配会使对象的可...


 Lu_gl [关注](#)

 11   59  1



Lu_gl
码龄5年

暂无认证

47 6万+ 6万+ 38万+ 
原创 周排名 总排名 访问 等级

1721 101 284 31 1385
积分 粉丝 获赞 评论 收藏



私信

关注

侯捷的C++大师系列课

立即领取 >

勿在浮沙筑高台

侯捷

两岸著名C++技术专家



搜博主文章



热门文章

向量点乘(即内积)和叉乘(即外积、向量积)
区别与意义分析  75381

excel计算出均方根值 (RMS) + 均方根误差 (RMSE) +标准差 (Standard Deviation)  65572

Matlab调整子图位置及大小  21196

C++中类的实例化的方法  18042

eigen库的安装与基本使用教程  15165

分类专栏



常用图形绘制

2篇



Lu_gl

关注

 11



 59

 1

最新评论

Ubuntu 18.04.3 双屏显示其中一个屏幕黑...
Mustrg: 根据您的操作建议，成功把两个屏幕都搞黑屏了

Linux/ubuntu build编译make时出现has ...
普通网友: 支持一下，细节很到位！【我也写了一些相关领域的文章，希望能够得到 ...

pdfLaTeX和XeLaTeX的图片无法识别与...
qq_47269923: 所以使用xelatex编译TEX文件，只能支持插入png格式的图片吗？

VS中出现模块计算机类型“X86”与目标计...
热爱机器视觉: 连接器--常规--链接库依赖项，选择否。

VS中出现模块计算机类型“X86”与目标计...
热爱机器视觉: 把可执行目录改成C:\Program Files\OpenCV4.8.0\opencv\build\x64' ...

最新文章

针对VMWare无法使用鼠标功能键问题

Ubuntu：解决github出现 Permission denied (publickey)的问题

关于Latex的引用网页、更改图表题样式、IEEEbiography 作者简介常规问题随笔

2024年 6篇	2023年 1篇
2022年 10篇	2021年 6篇
2020年 22篇	2019年 3篇

目录

C++中类的实例化的方法

- 一、隐式创建
- 二、显式创建



Lu_gl

关注

11



59

1

三、显式new创建



Lu_gl

关注



11



59



1