

为什么c++需要右值引用

原创 bitkevin 于 2021-10-15 23:50:36 发布 530 收藏 3 版权

分类专栏: c++ 文章标签: c++



C++ 专栏收录该内容

0 订阅 3 篇文章

订阅专栏

c++旧标准中有引用类型和指针类型，从c++11标准开始引入了右值引用，原来的引用被称之为左值引用，今天我们就来看一下右值引用是什么？右值引用是用来解决什么问题的？

1. 什么是右值引用

1.1左值和右值

c++的 **表达式** 可以是左值或者右值。左值和右值的概念是从c语言继承而来的，原本的含义是：左值可以放在赋值语句的左侧，右值则不能。

但是在c++中，左值和右值的区别没那么简单。例如常量表达式为代表的一些左值实际上不能作为赋值语句的左侧运算对象。

当一个对象被用作右值的时候，用的是对象的值(内容)；当对象被用作左值的时候，用的是对象本身(内存中的位置)。

需要使用左值的地方：

- 赋值运算符需要一个左值作为左侧运算对象，得到一个左值。
- 取地址符(&)作用左值运算对象，得到这个对象的指针(右值)。
- 解引用(*)和下标运算符[]求值结果为左值。
- 内置类型和迭代器的递增递减运算符(++,-)需要作用于左值，前置自增自减返回左值，后置自增自减返回右值。
- 字符串字面量如“hello world”

右值是没有标识符、不可以取地址的表达式，一般也称之为“临时对象”。最常见的情况有：

- 返回非引用类型的表达式，如 `x++`、`x + 1`、`make_shared(42)`
- 除字符串字面量之外的字面量，如 `42`、`true`

1.2左值引用和右值引用

在旧标准中的引用是对表达式的一个别名，c++11中将旧标准中的引用称之为左值引用，使用&表示。与其相对的是右值引用，只能绑定到右值，使用&&表示。右值引用只能绑定到一个即将销毁的变量。

2. 移动

在c++11中，虽然不能将一个右值引用绑定到一个左值上，但是可以显示地将一个左值转换为对应的右值引用。使用标准库提供的`std::move()`函数来实现，调用了`move`意味着这个左值除了销毁和重新赋值以为，我们不再使用它。

3. 右值引用的使用场景

3.1 移动构造函数

提供移动构造函数可以高效拷贝一些右值对象或不再使用的左值对象。

```
1 class TestDemo
2 {
3 public:
4     TestDemo(int num)
5     {
6         std::cout <<
```



bitkevin

关注

1 1 1

```

6     }
7     TestDemo(const TestDemo& other) :num(other.num) {
8         std::cout << "调用拷贝构造函数" << ", " << num << endl;
9     }
10    TestDemo(TestDemo&& other) noexcept :num(other.num) {
11        std::cout << "调用移动构造函数" << ", " << num << endl;
12    }
13 private:
14     int num;
15 };
16 int main() {
17     TestDemo test(1);
18     TestDemo test2(std::move(test));
19 }

```

运行输出

```

调用构造函数,1
调用移动构造函数,1

```

在上面的示例中调用构造函数构造了一个对象test, test对象如果在后面不再使用, 则可以通过std::move获取它的右值引用进而调用移动构造函数, 避免了数据的拷贝。

3.2 移动赋值

除了移动构造函数, 我们还可以定义移动赋值操作符, 可以将一个右值移动到左值上, 避免数据拷贝。

```

1     TestDemo &operator=(TestDemo&& other) noexcept {
2         if(this != &other) {
3             //todo::释放资源
4             num = other.num;
5         }
6         std::cout << "调用移动赋值" << endl;
7         return *this;
8     }
9     TestDemo test(1);
10    TestDemo test2(2);
11    test2 = std::move(test);

```

运行输出结果:

```

调用构造函数,1
调用构造函数,2
调用移动赋值

```

3.3 push_back

```

1 class TestDemo
2 {
3 public:
4     TestDemo(int num):num(num){
5         std::cout << "调用构造函数" << ", " << num << endl;
6     }
7     TestDemo(const TestDemo& other) :num(other.num) {
8         std::cout << "调用拷贝构造函数" << ", " << num << endl;
9     }
10    TestDemo(TestDemo&& other) noexcept :num(other.num) {
11        std::cout << "调用移动构造函数" << ", " << num << endl;
12    }
13 private:
14     int num;
15 };
16 int main() {
17     std::vector<Test

```



bitkevin

关注

1

```
18 |     TestDemo test{1};
19 |     demo1.push_back(test);
20 | }
```

初始化一个有两个元素的vector,然后再插入一个左值。我们知道vector超过容量之后要扩容然后拷贝原有数据。但如果类定义了一个**noexcept**的移动构造函数，原始数据将不用拷贝构造函数拷贝，而使用移动构造，大大提高效率。以上代码输出：

```
调用构造函数,2
调用构造函数,3
调用拷贝构造函数,2
调用拷贝构造函数,3
调用构造函数,1
调用拷贝构造函数,1
调用移动构造函数,3
调用移动构造函数,2
```

但是如果没有定义移动构造函数，那就需要拷贝了，如下示例：

```
1 | class TestDemo
2 | {
3 | public:
4 |     TestDemo(int num):num(num){
5 |         std::cout << "调用构造函数" <<"," << num << endl;
6 |     }
7 |     TestDemo(const TestDemo& other) :num(other.num) {
8 |         std::cout << "调用拷贝构造函数" <<"," << num << endl;
9 |     }
10 |    // TestDemo(TestDemo&& other) noexcept :num(other.num) {
11 |    //     std::cout << "调用移动构造函数" <<"," << num << endl;
12 |    // }
13 | private:
14 |     int num;
15 | };
16 | int main() {
17 |     std::vector<TestDemo> demo1{2,3};
18 |     TestDemo test{1};
19 |     demo1.push_back(test);
20 | }
```

运行输出：

```
调用构造函数,2
调用构造函数,3
调用拷贝构造函数,2
调用拷贝构造函数,3
调用构造函数,1
调用拷贝构造函数,1
调用拷贝构造函数,3
调用拷贝构造函数,2
```

c++11为什么要引入右值引用waxtear 的博客 3138

本文适合对c++11刚入门的小伙伴阅读。本文加深对左值引用、右值引用、左值右值的理解。

右值引用及其作用caojianfa969 的博客 1614

什么是左值、右值？可以出现在=号左边的则为左值，否则为右值。可以出现在=左边，意味...

C++:浅谈右值引用_青 nih 的博客6-10

为了支持移动操作,c++新标准引入了一种新的引用类型—右值引用。所谓右值引用就是必须绑...

现代C++ 理解右值引用_边缘计算机的博客_c++ 右值引用7-19

右值引用是C++11中新增新的引用类型。右值引用只不过是一种新的 C++ 语法。基于右值引...

C++ 右值引用的由来

与C++语言分手好多年，重识C++发

 bitkevin

关注

- C++——右值引用

yi_chengyu 的博客 1323

文章目录一、基本知识1.作用2.右值的基本概念以及与左值的辨别3.右值引用的特性二、右值...
- C++11的右值引用是什么意思？

Jan__ 的博客 369

什么是左值？右值？顾名思义，左值代表=左边的内容，可被修改，即非const变量都是左值...
- C++ 左值引用与右值引用的作用及实现原理

AI06le 的博客 480

定义及作用 左值：表示对象的身份，生命周期持久。对左值的修改实际上就是对对象的修改...
- 详解C++中右值引用

LXY 的博客 4599

98中的引用 概念 特性 引用的使用场景 三种传参方式效率的比较 探索:引用的底层实现方式---...
- C/C++编程：如何评价C++11的右值引用特性

OceanStar 的博客 101

右值引用是C++11中最重要的新特性之一，它解决了C++中大量的历史遗留问题，使C++标准...
- C++中的左值、右值、左值引用、右值引用

最新发布 计算机视觉的学习日常 432

C++中的左值、右值、左值引用、右值引用
- C++右值引用

番茄都是西红柿 152

链接：https://www.zhihu.com/question/22111546/answer/30801982 来源：知乎 右值引用是...
- 右值引用的作用

pkxpp 的专栏 7353

文章目录右值引用Move SemanticsPerfect Forwarding参考 右值引用 自从引入了这个右值引...
- C++ C++11新特性--右值引用

WhiteShirtI 的博客 325

左值与右值 在C语言中，左值和右值一般有两种区分的方法。可以出现在赋值符号"="的两边...
- C++ std::move原理&右值引用存在的必要性

2359Coding 1575

在学习C++的过程中一直搞不懂为什么要存在右值引用，将左值引用变为右值引用 个人认为...
- C++中左值（引用）及右值（引用）详解

一包辣条包邮！ 的博客 425

写C++代码编译时，有时会出现左值问题错误或右值错误，那左值和右值究竟是什么呢？？...
- 右值引用的由来

qq_39790009 的博客 433

https://www.bilibili.com/video/BV15v411g7Ma/?spm_id_from=autoNext 对这个视频的笔记，...
- C++右值引用的效果测试

丁劲舜技术发布空间 1043

右值引用主要作用是解决大对象在作为函数返回值返回时的深度拷贝问题，以及大对象之间的...
- 移动构造函数及右值左值引用详解

m0_48643365 的博客 123

1.移动构造函数：有时候我们会遇到这样一种情况，我们用对象a初始化对象b后对象a我们就...
- C++11 右值引用

zhongzh8 的博客 165

右值引用的目的主要是为了减少内存拷贝，优化性能。左值 右值 赋值操作符"="的左侧，...
- C++ 右值引用

weixin_43688483 的博客 326

文章目录1、右值引用介绍1.1 move函数2、右值作用 1、右值引用介绍 我们首先按确定什...
- C/C++编程：右值引用

OceanStar 的博客 1734

右值引用不过是C++的一种新语法，重要的是基于右值引用引申处理的两种C++编程技巧：移...

“相关推荐”对你有帮助么？

- 😞

非常没帮助
- 😐

没帮助
- 😬

一般
- 😄

有帮助
- 😂

非常有帮助

©2022 CSDN 皮肤主题：深蓝海洋 设计师：CSDN官方博客 返回首页

关于我们

招贤纳士

商务合作

寻求报道

400-660-0108

kefu@csdn.net

在线客服

工作时间 8:30-22:00

公安备案号11010502030143

京ICP备19004658号

京网文〔2020〕1039-165号

经营性网站备案信息

北京互联网违法和不良信息举报中心

家长监护

网络110报警服务

中国互联网举报中心

Chrome商店下载

©1999-2022北京创新乐知网络技术有限公司

版权与免责声明

版权申诉

出版物许可证

营业执照



bitkevin
码龄1年

暂无认证

24

原创

25万+

周排名

14万+

总排名

1万+

访问

等级

276

46

28

11

54



bitkevin

关注

1





私信

关注

搜博主文章



Website Templates For Any Idea

With Squarespace, you have access to a wide range of beautiful website templates.

Squarespace

Open

热门文章

- 常见路径规划算法介绍 5122
- 在IDEA/CLion/PyCharm中使用LeetCode插件详解 1300
- mac上使用clion编译调试c++程序 1272
- docker常用命令 771
- Envoy架构介绍与配置说明 706

分类专栏

	中间件	1篇
	学习笔记	5篇
	c++	3篇
	数据结构与算法	10篇
	效率工具	6篇
	网络	1篇



最新评论

在IDEA/CLion/PyCharm中使用LeetCode...
bitkevin: 得有工程文件，makefile或者cmake才能去掉。单独的leetcode题不属于工 ...



bitkevin

关注

1



蚂蚁的信息素: 博主，上面的黄色大横幅怎么去掉啊

二叉树的遍历(深度优先/广度优先 递归/...
bitkevin: 感谢支持

二叉树的遍历(深度优先/广度优先 递归/...
士别三日wyx: 学到了，三连支持一波👍
树/二叉树的层序遍历 掌握这个方法轻松...
士别三日wyx: 优秀优秀👍

您愿意向朋友推荐“博客详情页”吗？



强烈不推荐 不推荐 一般般 推荐 强烈推荐

最新文章

- Envoy架构介绍与配置说明
- 无聊与兴奋 《幸福之路》读书笔记
- STL标准库设计原理与使用

2022年 2篇 2021年 23篇

目录

- 1. 什么是右值引用
- 2. 移动
- 3. 右值引用的使用场景