链滴

remove_if 算法

作者：heyang5188
原文链接：https://ld246.com/article/1505467205210
来源网站：链滴
许可协议：署名-相同方式共享 4.0 国际 (CC BY-SA 4.0)

remove if

今天在实际生活中遇到了一个代码，开始感觉很奇怪，够来查阅了书籍之后发现了以前没有注意到的removeif是一个重排类型的算法，不会执行容器的操作

泛型算法不会执行容器的操作

```
#include <algorithm>
this->PowerUps.erase(std::remove_if(this->PowerUps.begin(), this->PowerUps.end(),
        [](const PowerUp &powerUp) { return powerUp.Destroyed && !powerUp.Activated; }
    ), this->PowerUps.end());
```

函数remove_if()移除序列[start, end)中所有应用于lambda表达式返回true的元素.

此函数返回一个指向被修剪的序列的最后一个元素迭代器.

实际上 remove_if()并不会实际移除序列[start, end)中的元素; 如果在一个容器上应用remove_if(), 容的长度并不会改变(remove_if()不可能仅通过迭代器改变容器的属性), 所有的元素都还在容器里面.

所以，我们的这里的是这样，显示用removeif函数将所有 powerup中的需要移除的元素全部移到容的后部分，

返回一个指向这个后面被移动了的函数的迭代器

然后调用erase函数将这些元素移除