

卖程序的小歪

博客园 | 首页 | 新随笔 | 联系 | 订阅 | 管理

随笔 - 453 文章 - 0 评论 - 5 阅读 - 19万

昵称： 卖程序的小歪
园龄： 7年9个月
粉丝： 18
关注： 1
+加关注

< 2021年12月 >						
日	一	二	三	四	五	六
28	29	30	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

搜索

找找看

谷歌搜索

常用链接

我的随笔
我的评论
我的参与
最新评论
我的标签

随笔分类 (464)

Algorithm(317)
C/C++(39)
GameDev(7)
Hadoop Ecosys(6)
Java(23)
Linux Common(37)
Linux Kernel(4)
MySQL(11)
Network(6)
Python(9)
Tools(4)
Web(1)

随笔档案 (453)


2017年5月(2)
2016年5月(1)
2015年11月(1)
2015年10月(2)
2015年9月(22)
2015年8月(19)
2015年7月(21)
2015年6月(26)
2015年5月(26)
2015年4月(16)
2015年3月(24)
2015年1月(9)

C++ STL: lower_bound与upper_bound实现


lower_bound

lower_bound(begin, end, target)用来查找一个已排序的序列中[begin, end)第一个大于等于target的元素index。数组A如下：
value: 1, 2, 2, 3, 4, 5, 5, 6, 7
index: 0, 1, 2, 3, 4, 5, 6, 7, 8


这样的序列，如果查找5的lower_bound，返回的应该是第一个5即A[5]。下面是摘自cplusplus.com上的lower_bound代码




```
template <class ForwardIterator, class T>
ForwardIterator lower_bound (ForwardIterator first, ForwardIterator last, const T& val)
{
    ForwardIterator it;
    iterator_traits<ForwardIterator>::difference_type count, step;
    count = distance(first, last);
    while (count > 0)
    {
        it = first; step = count / 2; advance (it, step);
        if (*it < val) { // or: if (comp(*it, val)), for version (2)
            first = ++it;
            count -= step + 1;
        }
        else count = step;
    }
    return first;
}
```

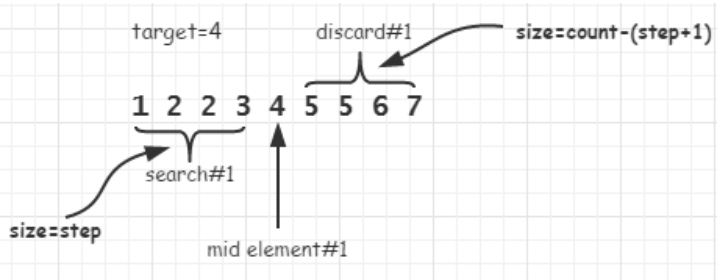


如果搜索对象只是数组的话还可以再简化一点：



```
count = last - start;
while (count > 0) {
    step = count / 2;
    int* it = first + step;
    if (*it < target) {
        count = count - (step + 1);
        first = it + 1;
    } else {
        count = step;
    }
}
return first;
```





[首页](#)[新闻](#)[博文](#)[专区](#)[闪存](#)[班级](#)[代码改变世界](#)[注册](#)[登录](#)

2014年10月(20)

[更多](#)

相册 (1)

[雪\(1\)](#)

阅读排行榜

1. CSRF & CORS(16661)
2. Java 远程调试(9040)
3. Maven 那些破事(7586)
4. Oracle JDBC 连接卡死后 Connection Reset(6267)
5. Linux 调试: systemtap(5878)

评论排行榜

1. OpenStack IceHouse 部署 - 5 - 网络节点部署(3)
2. PAT 1034. Head of a Gang[bug](1)
3. PAT 1010 Radix(1)

推荐排行榜

1. CSRF & CORS(10)
2. Java 远程调试(2)
3. 协程与Epoll的配合(1)
4. Offer选择与总结(1)
5. 字符串哈希函数(1)

最新评论

1. Re: PAT 1034. Head of a Gang[bug]
我也是遇到段错误，最后找到原因是G数组开小了，N最大为1000说明节点有可能有2000个

--PengHao's_Blog

2. Re: PAT 1010 Radix

请问最后一句：

“哎没办法，老女人就是爱这样。”
是啥意思

--GuanXi12

3. Re: OpenStack IceHouse 部署 - 5 - 网络节点部署

@ 卖程序的小歪多谢指导，我是按照，但是由于单网卡，参考了您的网络节点配置，看了您的pdf，发现我的没有默认的ext-net，是不是哪里遗漏了呢，多谢。...

--cutebupt

4. Re: OpenStack IceHouse 部署 - 5 - 网络节点部署

@ cutebupt配置文档的话还是看官方的比较好，我这个其实就是当时根据自己环境在官方文档上改的。使用文档的话我写过一个刚刚传到百度网盘上了：有问题邮件交流...

--卖程序的小歪

5. Re: OpenStack IceHouse 部署 - 5 - 网络节点部署

你好，我也是在学校机房，实验环境基本一样，按您的方法搭建之后，虚拟机还是上不了外网，原因可能是没有设置好floating ip，请问您有怎么设置网络启动虚拟机的文档吗，我q876602966，能交流一...

--cutebupt

当在上图中的数组中找4的lower_bound时，第一次*it取到的值是4，因为这不是简单的二分搜索，而是要返回第一大于等于查找元素的位置，所以搜索不能在此时结束。但是可以确定5~7这一部分可以不用搜索了，因为当前至少有一个元素即*it是大于等于4了，因而缩小查找范围（count=step）。这个查找范围并不包括已找到的4，为什么是这样？分情况讨论：

1. 当前面的这个范围没有符合条件的数时，就会将范围最后的位置的后一位置返回，而此位置正好是4所在的位置（*it = target时it所在的位置，它是符合查找条件的），其正好是lower_bound。

2. 当前面的这个方位含有符合条件的数时，此时当前的这个4就不是lower_bound，真正的lower_bound会在该区间内产生

case: target=5

当求5的lower_bound时，第一次找到中间元素时4，4<5，所以4和4前面的所有都不会含有5的lower_bound，因而下一次搜索只会在5~7这个区间进行，这个就和一个全新的问题一样了。

upper_bound

upper_bound用来在[begin, end)中找到第一个大于target的index



```
template <class ForwardIterator, class T>
ForwardIterator upper_bound (ForwardIterator first, ForwardIterator last, const T& val)
{
    ForwardIterator it;
    iterator_traits<ForwardIterator>::difference_type count, step;
    count = std::distance(first, last);
    while (count>0)
    {
        it = first; step=count/2; std::advance (it, step);
        if (! (val<*it)) // or: if (!comp(val,*it)), for version (2)
        { first=++it; count-=step+1; }
        else count=step;
    }
    return first;
}
```



简化版本：

lower_bound:



```
1      int lo = 0, hi = n;
2      // lower_bound
3      while (lo < hi) {
4          int mid = (lo + hi) / 2;
5          if (A[mid] < target) {
6              lo = mid + 1;
7          } else {
8              hi = mid;
9          }
10     }
11
12     return lo;
```



upper_bound:



```
lo = 0, hi = n;
// upper_bound
while (lo < hi) {
    int mid = (lo + hi) / 2;
    if (A[mid] <= target) {
        lo = mid + 1;
    }
}
```

```
        }  
    }  
    return lo;  
}
```

就在判断条件上多了个等号

分类: [Algorithm](#), [C/C++](#)

好文要顶

关注我

收藏该文







[卖程序的小歪](#)
关注 - 1
粉丝 - 18

[+加关注](#)

0

0

« 上一篇: [LeetCode Find Minimum In Rotated Sorted Array II](#)
» 下一篇: [微软2016校园招聘在线笔试-Professor Q's Software](#)

posted @ 2015-04-02 14:15 卖程序的小歪 阅读(1295) 评论(0) 编辑 收藏 举报

[刷新评论](#) [刷新页面](#) [返回顶部](#)

登录后才能查看或发表评论, 立即 [登录](#) 或者 [逛逛](#) 博客园首页

- 【推荐】字节跳动旗下火山引擎，邀您共同揭秘“云+增长”
- 【推荐】冬天里的一把火，园子里的一朵云：满园尽是火山引擎
- 【推荐】跨平台组态\工控\仿真\CAD 50万行C++源码全开放免费下载！
- 【推荐】华为 HarmonyOS 校园极客秀征文活动第1期，秀出你的COOL

编辑推荐:

- .NET Core 如何配置 TLS Cipher (套件) ?
- 记一次 .NET 某智能服装智造系统 内存泄漏分析
- 大学毕业三年的一些经历与思考
- 简单聊下 .NET6 Minimal API 的使用方式
- 使用 CSS 轻松实现一些高频出现的奇形怪状按钮



最新新闻:

- 超3成年轻人从未准点下班 加班最狠行业出炉：互联网第一 (2021-12-07 11:33)
 - 淘宝启动“2021十大年度商品”评选 搪瓷痰盂、小黄鸭发卡呼声最高 (2021-12-07 11:28)
 - 微软海洋环保鼠标今日上市：199元 一节电池用12个月 (2021-12-07 11:23)
 - 特斯拉撤回12亿美元补贴申请 德国经济部长：着急建厂 等不及审批 (2021-12-07 11:17)
 - 电子烟渠道或被重构 一周三次监管落地 (2021-12-07 11:10)
- » 更多新闻...