

与君共勉

未曾清贫难成人  
不经打击老天真  
自古英雄出炼狱  
从来富贵入凡尘！

欢迎关注微博  
个人邮箱：  
morewindows#126.com

个人资料



MoreWindows

访问：7081455次

积分：37085

等级：BLOG 8

排名：第117名

原创：156篇

转载：0篇

译文：0篇

评论：4987条

博客专栏



白话经典算法

文章：17篇

阅读：1918280



秒杀多线程面试题系列

文章：15篇

阅读：1327200



Windows C/C++/C# 编程

文章：130篇

阅读：6542289



C++ STL

文章：11篇

阅读：524390

阅读排行

白话经典算法系列之六 格

(457123)

【OpenCV入门指南】第

(435308)

白话经典算法系列之七 格

(361818)

白话经典算法系列之五 步

(292679)

秒杀多线程第二篇 多线程

(199610)

赠书 | 异步2周年,技术图书免费送 程序员8月书讯 项目管理+代码托管+文档协作，开发更流畅

白话经典算法系列之六 快速排序 快速搞定

标签： 微软 快速排序-快速搞定 算法 腾讯 面试

2011-08-13 17:19 457234人阅读 评论

分类： 白话经典算法系列 (15)

版权声明：本文为博主原创文章，未经博主允许不得转载。

快速排序由于排序效率在同为O(N\*logN)的几种排序方法中效率较高，因此经常被采用，再加上快速排序思想----分治法也确实实用，因此很多软件公司的笔试面试，包括像腾讯，微软等知名IT公司都喜欢考这个，还有大大小小的程序方面的考试如软考，考研中也常常出现快速排序的身影。

总的说来，要直接默写出快速排序还是有一定难度的，因为本人就自己的理解对快速排序作了下白话解释，希望对大家理解有帮助，达到快速排序，快速搞定。

快速排序是C.R.A.Hoare于1962年提出的一种划分交换排序。它采用了一种分治的策略，通常称其为分治法(Divide-and-ConquerMethod)。

该方法的基本思想是：

1. 先从数列中取出一个数作为基准数。

2. 分区过程，将比这个数大的数全放到它的右边，小于或等于它的数全放到它的左边。

3. 再对左右区间重复第二步，直到各区间只有一个数。

虽然快速排序称为分治法，但分治法这三个字显然无法很好的概括快速排序的全部步骤。因此我的对快速排序作了进一步的说明：挖坑填数+分治法：

先来看实例吧，定义下面再给出（最好能用自己的话来总结定义，这样对实现代码会有帮助）。

以一个数组作为示例，取区间第一个数为基准数。

0	1	2	3	4	5	6	7	8	9	
72	6	5	7	88	60	42	83	73	48	85

初始时，i = 0; j = 9; X = a[i] = 72

由于已经将a[0]中的数保存到X中，可以理解成在数组a[0]上挖了个坑，可以将其它数据填充到这来。

http://blog.csdn.net/morewindows/article/details/6684558

1/12

秒杀多线程第四篇 一个线程 (171498)  
白话经典算法系列之一 冒泡排序 (170757)  
秒杀多线程第一篇 多线程 (163761)  
白话经典算法系列之三 希尔排序 (159501)  
【OpenCV入门指南】第... (136674)

MoreWindows微博



时光荏苒，个人技术博客就快要开博5周年了，承蒙大家厚爱，访问量日渐增加，如今马上就要到500万了，评论条数也达到了4612条。从青涩到成熟，从迷茫到坚定，5年走过也终于找到了自己愿意奋斗一生的事业！愿大家永远永远永远不要放弃对梦想的追求，长夜终将破晓、阳光终将照耀、而你将全力奔跑！



2016-7-10 22:52

转发了 [华尔街日报中文网](#) 的微博：【深度：中国中小企业融资难问题依然无

[登录](#) | [注册](#)

文章分类

白话经典算法系列 (16)  
Windows多线程 (15)  
STL 他山之石 (11)  
C/C++/C#基础 (18)  
HTML/javascript/PHP (12)  
Linux编程 (1)  
MoreWindows工作笔记 (12)  
Windows界面编程 (13)  
Windows编程 (87)  
VC6.0及VS2008使用技巧 (7)  
OpenCV入门指南 (13)  
生活 (1)

评论排行

白话经典算法系列之六 快速排序 (301)  
白话经典算法系列之七 堆排序 (196)  
秒杀多线程第六篇 经典归并排序 (174)  
白话经典算法系列之五 归并排序 (157)  
位操作基础篇之位操作全解 (148)  
秒杀多线程第三篇 原子操作 (133)  
秒杀多线程第五篇 经典归并排序 (132)  
秒杀多线程第十篇 生产者-消费者问题 (127)  
秒杀多线程第二篇 多线程 (123)  
STL系列之十 全排列(百练) (120)

从j开始向前找一个比X小或等于X的数。当j=8，符合条件，将a[8]挖出再填到上一个坑a[0]中。a[0]=a[8]; i++; 这样一个坑a[0]就被搞定了，但又形成了一个新坑a[8]，这怎么办了？简单，再找数字来填a[8]这个坑。这次从i开始向后找一个大于X的数，当i=3，符合条件，将a[3]挖出再填到上一个坑中a[8]=a[3]; j--;

数组变为：

0	1	2	3	4	5	6	7	8	9
48	6	57	88	60	42	83	73	88	85

i = 3; j = 7; X=72

再重复上面的步骤，先从后向前找，再从前向后找。

从j开始向前找，当j=5，符合条件，将a[5]挖出填到上一个坑中，a[3] = a[5]; i++;

从i开始向后找，当i=5时，由于i==j退出。

此时，i = j = 5，而a[5]刚好又是上次挖的坑，因此将X填入a[5]。

数组变为：

0	1	2	3	4	5	6	7	8	9
48	6	57	42	60	72	83	73	88	85

可以看出a[5]前面的数字都小于它，a[5]后面的数字都大于它。因此再对a[0...4]和a[6...9]这二个区间重复上述步骤就可以了。

对挖坑填数进行总结

1. i = L; j = R; 将基准数挖出形成第一个坑a[i]。
2. j--由后向前找比它小的数，找到后挖出此数填前一个坑a[i]中。
3. i++由前向后找比它大的数，找到后也挖出此数填到前一个坑a[i]中。
4. 再重复执行2，3二步，直到i==j，将基准数填入a[i]中。

照着这个总结很容易实现挖坑填数的代码：

```
[cpp]
01. int AdjustArray(int s[], int l, int r) //返回调整后基准数的位置
02. {
03.     int i = l, j = r;
04.     int x = s[l]; //s[l]即s[i]就是第一个坑
05.     while (i < j)
06.     {
07.         // 从右向左找小于x的数来填s[i]
08.         while(i < j && s[j] >= x)
09.             j--;
10.         if(i < j)
11.         {
12.             s[i] = s[j]; //将s[j]填到s[i]中，s[j]就形成了一个新的坑
13.             i++;
14.         }
15.     }
```

文章搜索

最新评论

【OpenCV入门指南】第十三篇  
Willing110: VS2010版本的XML  
分类器不能用的这个可以下, 感谢楼主<http://download.csdn....>

白话经典算法系列之七 堆与堆排  
minqianqi: @sayomin:确实是取  
左右键值小的那个, 博主这块写的有问题;

白话经典算法系列之七 堆与堆排  
minqianqi: @luop911123:你说的  
对, 是父节点的键值和左右子节  
点的键值比, 谁小取谁, 博主写的有问题;

白话经典算法系列之六 快速排序  
hbdxzyb: @dps006:经过验证是  
没问题的, 你确定你的代码和博  
主的一样?

白话经典算法系列之六 快速排序  
shichunzao: 不错不错 书没看懂  
就搜到你的 看了 边算一下 就明  
白多了 深刻多了 谢谢啊 学习了

Windows系统CPU内存网络性能:  
szq2k08: 疑惑 高位时间为什么  
要乘以 4.294967296E9

Base64系列第三篇 C/C++中使用  
manonvsky: 楼主有独立思考过  
吗, 二进制数中有好多0值的,  
你用string做输入参数合适吗?

【白话经典算法系列之十四】腾i  
chenciyan: 涨见识了

秒杀多线程第一篇 多线程笔试面  
不会架构的厨师不是好司机: 来  
我的分布式群吗? 542241455

进程通信之二 管道技术第二篇 匿  
kewencommon: @huxi0328:我  
也发现了这个问题, , 我查了一  
下, 应该是子进程的printf有缓冲  
区, 等子进程结...

```

16.         // 从左向右找大于或等于x的数来填s[j]
17.         while(i < j && s[i] < x)
18.             i++;
19.         if(i < j)
20.         {
21.             s[j] = s[i]; //将s[i]填到s[j]中, s[i]就形成了一个新的坑
22.             j--;
23.         }
24.     }
25.     //退出时, i等于j。将x填到这个坑中。
26.     s[i] = x;
27.
28.     return i;
29. }

```

再写分治法的代码:

```

[cpp]
01. void quick_sort1(int s[], int l, int r)
02. {
03.     if (l < r)
04.     {
05.         int i = AdjustArray(s, l, r); //先成挖坑填数法调整s[]
06.         quick_sort1(s, l, i - 1); // 递归调用
07.         quick_sort1(s, i + 1, r);
08.     }
09. }

```

这样的代码显然不够简洁, 对其组合整理下:

```

[cpp]
01. //快速排序
02. void quick_sort(int s[], int l, int r)
03. {
04.     if (l < r)
05.     {
06.         //Swap(s[l], s[(l + r) / 2]); //将中间的这个数和第一个数交换 参见注1
07.         int i = l, j = r, x = s[l];
08.         while (i < j)
09.         {
10.             while(i < j && s[j] >= x) // 从右向左找第一个小于x的数
11.                 j--;
12.             if(i < j)
13.                 s[i++] = s[j];
14.
15.             while(i < j && s[i] < x) // 从左向右找第一个大于等于x的数
16.                 i++;
17.             if(i < j)
18.                 s[j--] = s[i];
19.         }
20.         s[i] = x;
21.         quick_sort(s, l, i - 1); // 递归调用
22.         quick_sort(s, i + 1, r);
23.     }
24. }

```

快速排序还有很多改进版本, 如随机选择基准数, 区间内数据较少时直接用另的方法排序以减小递归深度。有兴趣的筒子可以再深入的研究下。

注1, 有的书上是以中间的数作为基准数的, 要实现这个方便非常方便, 直接将中间的数和第一个数进行交换就可以了。

转载请标明出处, 原文地址: <http://blog.csdn.net/morewindows/article/details/6684558>

上一篇 VC 如何使用程序运行后自己删除自己


下一篇 使用VC库函数中的快速排序函数

相关文章推荐


- 移植linux内核(3.0.13)的链表实现进windows下，...
  - 【直播】70天软考冲刺计划--任钰
  - qsort 快排算法讲解
  - 【直播】打通Linux脉络 进程、线程、调度--宋宝华
  - 白话经典算法系列之一 冒泡排序的三种实现
  - 【直播】机器学习之凸优化--马博士
  - 基础算法之一--快排
  - 【套餐】MATLAB基础+MATLAB数据分析与统计-...
- 开博了
  - 【课程】3小时掌握Docker最佳实战--徐西
  - 【获奖公布】征文 | 你会为 AI 转型么？
  - 【课程】深度学习基础与TensorFlow实践
  - 快排算法及常见两种常见优化方法
  - 快排算法及利用快排思想寻找第k大（小）的数
  - 【算法】5 传说中的快排是怎样的，附实现
  - 快排算法

查看评论

78楼 [shichunzao](#) 3天前 19:32发表

 不错不错 书没看懂 就搜到你的 看了 边算一下 就明白多了 深刻多了 谢谢啊 学习了


77楼 [mu羊](#) 2017-08-10 15:53发表




```
public void QuickSort(int[] s, int l, int r)
{
    if (l <= r)
    {
        int i = l, j = r, x = s[l];
        while (i <= j)
        {
            while (i <= j && s[j] >= x)
                j--;
            s[i] = s[j];
            while (i <= j && s[i] <= x)
                i++;
            s[j] = s[i];
        }
        s[i] = x;
        QuickSort(s, l, i - 1);
        QuickSort(s, i + 1, r);
    }
}
```

不知道我现在回算不算洛阳铲，但我觉得吧，就每次换完还要再++，太多余了，而且这个++必须被i<j限制，不然还会出错


Re: [霖宝包](#) 前天 18:37发表

 回复qq\_35528646: ++用来减少后面while循环的次数，这里++了，后面while循环就少一次。这里要是不++，while循环就正常次数。

76楼 [dps006](#) 2017-08-08 15:42发表

 {9, 5, 1, 2, 6, 7, 3, 4, 8} 用这串数字验证5的位置不对啊，结果是1,2,3,4,6,7, 5 ,8,9


Re: [hbdxzyb](#) 3天前 09:21发表

 回复dps006: 经过验证是没问题的，你确定你的代码和博主的一样？

75楼 [DD\\_Davina](#) 2017-08-06 19:30发表

 非常好的讲解

74楼 [wang0112233](#) 2017-07-30 21:23发表

 请问如果说你交换到中间的时候，没有空余的位子给基数了，怎么解决  
如6,1,2,5,9,4,7,10,8

4和9进行交换以后，谁和6进行交换呢

Re: 庄志锋 2017-08-04 16:45发表



回复wang0112233: 4怎么会和9交换呢？4在第一轮j移动后就填到的那个坑里了，第一轮j移动后是这样的：4 1 2 5 9 坑 7 10 8，此时i指向4，j指向坑，第一轮i移动后是这样的：4 1 2 5 坑 9 7 10 8，此时i指向坑，j指向9，第二轮j移动就到了i的位置，此时就不用继续了，直接把基数填入坑内，最终结果是：4 1 2 5 6 9 7 10 8

Re: wang0112233 2017-08-09 10:06发表



回复mynameis121: 你说的是以中间一个数为基数，但是以最左边这个好像会遇到这个问题

73楼 Lazy\_Die 2017-07-15 10:12发表



楼主太强了，数据结构的算法都看你的，看一遍就懂。实在是太佩服了，帮我节省了好多时间。

72楼 hqfok 2017-07-12 15:09发表



楼主写的非常好，自卸写了一遍也转载了。感谢楼主分享

71楼 在京奋斗者 2017-07-10 22:53发表



楼主简直就是真神啊，顶礼膜拜。

70楼 是谁杀了爱丽丝 2017-06-16 17:56发表



溜得不行，几行就看懂了，比百度百科高到不知道哪里去了

69楼 aaaaa2428572 2017-06-05 10:36发表



服了大神求抱大腿

68楼 zhengyutou 2017-05-26 08:26发表



非常感谢楼主的分享，但是这个逻辑有个缺陷，当第一个数是数组中的最小值时，逻辑不成立。

Re: ctylovelxy 2017-07-09 11:36发表



回复zhengyutou: 如果是最小的，那就是从末尾一直比较到最前面，没问题啊

67楼 zhangxinguo201w 2017-04-08 17:02发表



按照你这样写，如何以第一个数为基准，那第一个数和最后一个数相等该怎么办？

Re: 忧郁的蛋糕 2017-07-29 13:05发表



回复zhangxinguo201w: 相等也得拿过来补到原来的坑中啊！  
这条代码while(i &lt; j &amp;& s[j] &gt;= x)  
就是包括了相等也要拿过去补坑的。

66楼 cy1534905480 2017-04-07 11:13发表



谢谢 楼主大大  
以前不懂 的地方今天终于搞懂了！！

65楼 gaoyu10happy 2017-04-04 19:38发表



楼主的代码有bug，quick\_sort(array,left,i-1);前需要判断 if (i > left) 否则代码会陷入死循环。

Re: 忧郁的蛋糕 2017-07-29 13:02发表



回复gaoyu10happy: 不会死循环吧，以为函数内部有if判断 左边要小于右边了。  
[code=cpp]  
void quickSort(arr, l, r) {  
 if (l &lt; r) {  
 while(xx)  
 xxx  
 quickSort(arr, l, i-1);  
 quickSort(arr, i+1, r);  
 }  
}

64楼 骨灰级菜鸟-程序猿 2017-03-29 09:52发表



按照楼主的意见明白了，  
看代码还要一点一点琢磨，一点一点套，  
不过代码看着好像有点问题：  
嵌套中的第一个循环好像比较时应该是<=把？

第二个循环是不是应该是>=啊

里面嵌套的两个循环是不是都缺少大括号啊???

63楼 [SoCleverbaby](#) 2017-03-26 19:33发表



```
while (i < j)
{
    while(i < j && s[j] >= x)
        j--;
    if(i < j)
    {
        s[i] = s[j];
        i++;
    }
}
```

不太懂上面大的循环中已经有i<j的条件了，下面的while为什么还有i<j的条件，还有交换之前也有这个条件下

Re: [li\\_xunfang](#) 2017-07-11 20:42发表



回复licongcong\_no1: 进去循环之后要一直保证i小于j才能有意义，最外面的while条件只在进入循环时判断。

62楼 [墨染百城](#) 2017-03-23 11:45发表



谢谢博主分享，一下就看懂了。以后博客写作要向博主多多学习！

61楼 [ice\\_martin](#) 2017-03-21 10:54发表



好文，原来我一直理解的不够深入

60楼 [Lin\\_YaoT](#) 2017-03-04 11:42发表



那两个递归是怎么退出的???

59楼 [TrainspottingAlan](#) 2017-03-02 13:16发表



厉害了，我的哥

58楼 [as](#) 2017-02-22 17:32发表



挖坑填坑对于理解代码有好处

57楼 [eddielyc](#) 2017-02-13 19:33发表



相见恨晚，讲的非常清楚，真的好喜欢这种风格的讲解方式

56楼 [qq\\_37420585](#) 2017-01-28 12:08发表



我觉得这样理解更好 从后面找一个比它小的数 从前面找一个比它大的数 把这两个数交换位置 对吗

55楼 [qq\\_37217771](#) 2016-12-31 11:24发表



谢谢老师！

54楼 [路漫远吾求索](#) 2016-12-28 10:29发表



调用时，如果数组中有相等的元素，里面的判断条件要改成>=以及<=才行

53楼 [AC\\_Dreameng](#) 2016-12-20 22:05发表



调用的时候要注意啊，最后一个参数为数组最后一个一个元素的下标，而不是数组长度!!!! 这个地方错了好久

52楼 [Mr\\_厚厚](#) 2016-12-18 16:53发表



再次验证，多谢

51楼 [DEEPPDARKVANTASY](#) 2016-12-05 12:45发表



挖坑很形象

50楼 [Three\\_\\_Years](#) 2016-12-03 20:32发表



i与j的作用是啥啊？楼主，求解

49楼 [denganliang825](#) 2016-11-12 18:55发表

厉害了word哥。



48楼 [Kazama\\_Kenji](#) 2016-11-02 21:59发表



浅显易懂，看得好爽啊

47楼 [meluobote](#) 2016-10-28 19:11发表



万分感谢，好文章，过了几个小时才懂。。。。。

46楼 [-Mickey-](#) 2016-10-28 13:49发表



博主您这两个地方“从j开始向前找一个比X小或等于X的数”和“这次从i开始向后找一个大于X的数”。写错了。是比x小的数，第二个向后找的是第一个大于等于x的数。

45楼 [alexxu1988](#) 2016-10-12 09:27发表



赞，挖坑的说法十分形象，秒懂

44楼 [qq\\_20563349](#) 2016-10-04 09:09发表



太棒了，理解的太透彻了

43楼 [Bboy-AJ](#) 2016-10-01 17:46发表



清晰明了。

42楼 [qq\\_35654343](#) 2016-09-28 17:40发表



超级感谢

41楼 [qq\\_36155671](#) 2016-09-16 23:47发表



while 下面没打括号 看晕我了

40楼 [\\_蓦然回首\\_](#) 2016-09-10 21:17发表



相见恨晚，醍醐灌顶！

39楼 [rqy1994](#) 2016-09-04 14:39发表



太棒了，浅显易懂。很不错。

38楼 [newmaomao](#) 2016-08-11 16:40发表



```
[cpp]
01.  /**
02.   * @author TIAN
03.   * @brief  交换数组位置i和位置j的元素
04.   */
05.  static void array_swap(int array[], int i, int j)
06.  {
07.      int temp = array[i];
08.      array[i] = array[j];
09.      array[j] = temp;
10.  }
11.
12.  /**
13.   * @author TIAN
14.   * @brief  快速排序（哨兵筛选部分）
15.   */
16.  static int array_qsort_partition(int array[], int low, int high)
17.  {
18.      int pivot = low;
19.
20.      while (low < high)
21.      {
22.          while (pivot < high)
23.          {
24.              if (array[pivot] <= array[high])
25.              {
26.                  high--;
27.              }
28.              else
29.              {
```

```

30.         array_swap(array, pivot, hight);
31.         pivot = hight;
32.         break;
33.     }
34. }
35.
36. while (low < pivot)
37. {
38.     if (array[low] <= array[pivot])
39.     {
40.         low++;
41.     }
42.     else
43.     {
44.         array_swap(array, low, pivot);
45.         pivot = low;
46.         break;
47.     }
48. }
49. }
50.
51. return pivot;
52. }
53.
54. /**
55.  * @author TIAN
56.  * @brief 快速排序（递归部分）
57.  */
58. static void array_qsort(int array[], int low, int high)
59. {
60.     if (low < high)
61.     {
62.         int pivot = array_qsort_partition(array, low, high);
63.         array_qsort(array, low, pivot - 1);
64.         array_qsort(array, pivot + 1, high);
65.     }
66. }

```

Re: 系解 2016-10-07 15:58发表



回复newmaomao: 可以

37楼 风稻妞 2016-08-11 11:10发表



建议楼主出书呀，出书，出书！！看现在的大学教程书讲的数据结构，简直看不懂呀

36楼 kurt17 2016-08-05 17:29发表



感谢楼主。终于看懂了。目前为止发现的真正负责任的把问题解释清楚的文章。

35楼 zuojian1992 2016-08-01 20:46发表



您好，关于“挖坑”， $n[i++] = n[j]$ ;  $n[j--] = n[i]$ ;  $n[i] = tmp$ ; 可以使用 $n[i]$ 和 $n[j]$ 交换来代替吗？代码如下

```

[Java]

01. public static void _quickSort(int[] n, int x, int y) { //快速排序
02.     int i = x, j = y;
03.     int tmp = n[x]; //中轴
04.     while(i < j) {
05.         while(i < j && tmp < n[j])
06.             j--;
07.         swap(n, i, j);
08.         /*if(i < j)
09.             n[i++] = n[j];*/
10.         while(i < j && tmp > n[i])
11.             i++;
12.         swap(n, i, j);
13.         /*if (i < j)
14.             n[j--] = n[i];*/
15.     }
16.     //n[i] = tmp;
17.     _quickSort(n, x, i - 1);
18.     _quickSort(n, i + 1, y);
19. }
20.
21. public static void quickSort(int[] n) { //快速排序
22.     _quickSort(n, 0, n.length - 1);

```



```
23.
24.     }
25.
26.     public static void swap (int[] n, int x, int y) {
27.         int temp = n[x];
28.         n[x] = n[y];
29.         n[y] = temp;
30.     }
```

34楼 李勇杰Charles 2016-07-24 20:00发表



这是我第一次看懂快速排序，非常感谢

33楼 鲁峰2012 2016-07-16 19:27发表



c++中数组元素是0开头的，这个博客的第一个元素按的是1，这样待排数组中的第一个元素就不会参与排序的，问什么大家都看不到？

Re: qq\_21041195 2016-08-05 16:46发表



回复luzubodfgs: 那应该是left和right的 L 和R

Re: JazOh 2016-07-17 12:46发表



回复luzubodfgs: l? 1?

Re: 鲁峰2012 2016-08-13 20:07发表



回复u010111321: l和1太像了 第一遍看 当成1了，建议l用别的数替代吧，

32楼 龙井88 2016-07-16 11:06发表



博主，我用php语言实现了，返回不了东西

```
function quick_sort(&$arr,$left,$right)
{
    if($left<$right)
    {
        $x=$arr[$left];
        $r=$right;
        $l=$left;
        while($l<$r&&$x<=$arr[$r])
        {
            $r--;
        }
        if($l<$r)
        {
            $arr[$l+]=$arr[$r];
        }
        while($l<$r&&$x>$arr[$l])
        {
            $l++;
        }
        if($l<$r)
        {
            $arr[$r-]=$arr[$l];
        }
        $arr[$l]=$x;
        if($l==$r)
        {
            quick_sort($arr,$left,$l-1);
            quick_sort($arr,$l+1,$right);
        }
    }
    else {return json_encode($arr);}
}
```

31楼 ZCShouCSDN 2016-07-10 16:58发表



支持博主，确实好理解不少！！

30楼 wzyorigin08 2016-07-10 13:06发表



[cpp]

```

01. [cpp] view plain copy
02. 01.//快速排序
03. 02.void quick_sort(int s[], int l, int r)
04. 03.{
05. 04.    if (l < r)
06. 05.    {
07. 06.        //Swap(s[l], s[(l + r) / 2]); //将中间的这个数和第一个数交换 参见注1
08. 07.        int i = l, j = r, x = s[l];
09. 08.        while (i < j)
10. 09.        {
11. 10.            while(i < j && s[j] >= x) // 从右向左找第一个小于x的数
12. 11.                j--;
13. 12.            if(i < j)
14. 13.                s[i++] = s[j];
15. 14.            while(i < j && s[i] < x) // 从左向右找第一个大于等于x的数
16. 15.                i++;
17. 16.            if(i < j)
18. 17.                s[j--] = s[i];
19. 18.        }
20. 19.    }
21.
22.
23.
24.    s[i] = x;
25.    // 这个地方有点不懂 这里为什么 不能把值 s[j] = x, 为什么只能赋值 给s[i] =x , 不
    等的吗
26.
27.
28.
29.    quick_sort(s, l, i - 1); // 递归调用
30.    quick_sort(s, i + 1, r);
31. }
32. }

```

Re: ZCShouCSDN 2016-07-10 18:29发表



回复wyzorigin08: 你测试不行? 测试数列多少

Re: tnoblew 2016-08-25 17:44发表



回复ZCShouCSDN: 我测试了下, x赋值给i或j都可以

29楼 woshinai 2016-07-01 20:17发表



还在最后把每一次比较和递归分开展示, 理解了。

28楼 u010749540 2016-06-24 10:38发表



博主写的很清晰易懂, 只是感觉代码风格不太好, 变量名i, j, l, r如果能改成left, right这种有意义的名称会更好。

27楼 THExChaos 2016-06-11 16:59发表



解决了, 是我没看自信, 没用while (l<j) 把两个while包起来

26楼 THExChaos 2016-06-11 16:54发表



有点问题, 我在Java里用博主的代码运行时候, 会出现数组第0, 1位被固定了的情况, 我采用的数组是{ 5, 8, 9, 7, 6, 3, 1, 2, 0, 4, 10 };目前正在思考解决方法中

25楼 awesomels 2016-06-04 09:37发表



学习了, 通俗易懂, 感谢感谢!

24楼 nearbyYoung 2016-06-02 22:17发表



看了几遍, 也终于看懂了

23楼 baidu\_17198579 2016-05-27 23:52发表



好文章, 程序通过。

22楼 我想我还有梦 2016-05-26 15:43发表



确实通俗易懂, 好文章。

21楼 薛瑄 2016-05-16 16:20发表



不得不说，真实通俗易懂，醍醐灌顶，好

20楼 [crev](#) 2016-05-15 20:33发表



太棒了，谢谢博主

19楼 [wushupeng3336](#) 2016-05-12 16:06发表



看了博主的解释，我一下就理解了快速排序，写的太棒了。给你一百个赞！！！！

18楼 [chensonglu](#) 2016-05-11 15:20发表



真的是很好理解，看了很多资料都很晕，这篇博客一看就全明白了

17楼 [游客] 2016-05-11 15:18发表



看了很多资料，这篇才是最简单易懂的

16楼 [deaolu164](#) 2016-04-27 23:04发表



非常感谢博主，写的很明白易懂！

15楼 [qq\\_33087307](#) 2016-04-13 13:29发表



我想问下为什么每次填数的时候要  $i++$ ， $j--$ 呢（ $s[i++] = s[j];$ ）

我尝试了下把自加自减去掉，发现会出现错误。但是去掉后应该不是就在下一次  $i$  或者  $j$  的循环里多循环一次吗？想不通为什么不能去掉

Re: [duckcaptain](#) 2017-06-30 09:27发表



回复 [qq\\_33087307](#): 如果保证数是不重复的，是可以，如果有重复数会陷入死循环原因在于，

```
[java]
01.  if (i < j) {
02.      s[i] = s[j];
03.      // i++;
04.  }
05.  while (i < j && s[i] < x) {
06.      i++;
07.  }
```

如果去掉  $i++$ ， $s[i]$  可能会等于  $x$  导致后续的  $i++$  无法继续推进。知道这点之后，所以不然就选择  $i++$ ，或者修改为  $\&\& i =$  都是可以的。

14楼 [千山牧雪](#) 2016-04-05 20:17发表



感谢楼主的讲解，认真研读你所讲的“填坑分治”法之后，我也写出了自己的快速排序，谢谢拜谢

13楼 [liteblue](#) 2016-04-01 11:35发表



写得真好，很容易就明白了，谢谢楼主

12楼 [kmcfly](#) 2016-03-24 20:01发表



写的对。。

11楼 [liuqian00070](#) 2016-03-21 11:21发表



你好，在从左到右找比  $x$  大的循环中： $\text{while}(i < j \ \&\& \ s[i] < x)$ ，为什么不用  $\text{while}(i < j \ \&\& \ s[i] \leq x)$ ，这样难道不是减少一次复制操作，效率更优吗？

10楼 [LOLERSB](#) 2016-03-16 15:08发表



总算理解了快排的方法，谢谢楼主

9楼 [tingxuelouwq](#) 2016-03-14 22:24发表



楼主，有个问题想请教下。

您的程序  $\text{while}$  循环：

$\text{if}(i < j)$

$s[i++] = s[j];$

中的  $\text{if}(i < j)$  这个判断意义在哪里呀？直接  $s[i++] = s[j]$  有什么问题吗？

Re: [tingxuelouwq](#) 2016-03-14 23:35发表

回复 [tingxuelouwq](#): 额，知道了为什么了，如果没有找到，则  $\text{left} = \text{right}$ ，因此需要进行判断



8楼 的得德得 2016-03-02 19:43发表



int x = s[i]; //s[i]即s[i]就是第一个坑  
小白问题：这里不应该是s[0]是第一个坑吗？楼主这样不是s[1]成为第一个坑，那s[0]没有比到哎

Re: crev 2016-05-15 22:02发表



回复go\_liyang：是字母L(大写方便识别) 不是数字1

Re: 看不清的天 2016-03-02 21:27发表



回复go\_liyang：s[0]是第一个坑，把这个数拿出和后面的数一一作比较。

7楼 aiterator 2016-02-03 22:58发表



博主写的太棒了！

6楼 heartacker 2016-02-02 12:44发表



小白想问一下。输入的int l 和int r 输入的是怎么值？

Re: 看不清的天 2016-03-02 21:23发表



回复heartacker：数组中的第一个数和最后一个数，通常为0，n-1；

5楼 大号小白兔 2016-01-25 09:44发表



大赞博主，传道授业解惑

4楼 wssy213 2015-12-28 11:03发表



如果去掉这些条件，将会导致 j 指向错误的位置；还可能导致数组访问越界。  
比如，将里面两个 while 循环中的 i < j 条件去掉后，使用 3,1 这两个数据来测试，看看会是怎样的结果。

Re: wssy213 2015-12-28 11:04发表



回复u013350333：我这是回复 157 楼的...

3楼 xiaolu091 2015-12-27 23:13发表



讲得很好

2楼 mingyunyuansu 2015-12-23 22:10发表



严蔚敏的书看得一头雾水。。。反正我是看博主的秒懂，还有其他的白话系列也非常清楚，学习一个

1楼 newbie\_ten 2015-12-19 23:52发表



太棒了！是我这菜鸟想要的

查看更多评论

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

