**[快速排序里的学问：枢纽元选择与算法效率](http://www.nowamagic.net/librarys/veda/detail/2397)**

**选择首尾元素做枢纽元**

通常的、没有经过充分考虑的选择是将第一个或最后一个元素用作枢纽元。选择第一个元素作为枢纽元的程序例子可以参考专题的前一篇《[快速排序里的学问：霍尔快排的实现](http://www.nowamagic.net/librarys/veda/detail/2396)》，而选择最后一个元素用作枢纽元的程序例子则可以参考《[快速排序里的学问：快速排序的过程](http://www.nowamagic.net/librarys/veda/detail/2390)》这个算法导论里的例子。

选择最后一个元素作为枢纽元的排序过程是这样的：



如果输入是随机的，那么这是可以接受的，但是如果输入是预排序的或者是反序的，那么这样的枢纽元就产生一个劣质的分割，因为所有的元素不是被划入S1就是被划入S2。更有甚者，这种情况发生在所有的递归调用中。

实际上，如果第一个元素用作枢纽元而且输入是预先排序的，那么快速排序所花费的时间将是二次的。然而，预排序的输入（或者有一大段预排序数据的输入）也是相当常见的，因此，使用第一个元素作为枢纽元的[算法效率](http://www.nowamagic.net/librarys/veda/tag/%E7%AE%97%E6%B3%95%E6%95%88%E7%8E%87)不是很高的。

另一种想法是选取前两个互异的键中的较大者作为枢纽元，但这和只选取第一个元素作为枢纽元效率也差不多。

**随机选取枢纽元**

一种安全的方针是随机选取枢纽元。

一般来说这种策略非常安全，除非随机数生成器有问题，因为随机的枢纽元不可能总在接连不断地产生劣质的分割。另一方面，随机数的生成一般是昂贵的，根本减少不了算法其余部分的平均运行时间。

**三数中分割法**

一组N个数的中值是第[N/2]个最大的数。枢纽元的最好的选择是数组的中值。

可是，这很难算出来，并且会明显减慢快速排序的速度。这样的中值的估计可以通过随机选取三个元素并用它们的中值作为枢纽元而得到。事实上，随机性并没有多大的帮助，因此一般的做法是使用左端、右端和中心位置上的三个元素的中值作为枢纽元。显然使用三数中值分割法消除了预排序输入的不好情形。

这种[枢纽元](http://www.nowamagic.net/librarys/veda/tag/%E6%9E%A2%E7%BA%BD%E5%85%83)选择的的快排，效率可是相当高的。

快速排序还很有很多各种变种，我们这里只研究几个有代表性的算法。