**[快速排序里的学问：霍尔与快速排序](http://www.nowamagic.net/librarys/veda/detail/2391)**

上一篇介绍了排序的本质，还有实现了《算法导论》里的快速排序算法。但是快速排序的算法不是只有一个，我们要一次过把快速排序的好东西都挖掘出来。所以这篇文章，让我们对快速排序溯源，去了解快速排序算法的发明者。



霍尔(Hoare)

霍尔 (Sir Charles Antony Richard Hoare) 是一位英国计算机科学家，他是著名的快速排序 (QuickSort) 的发明者。在平均状况下，排序 n 个项目要Ο(n log n) 次比较，而且通常明显比其他Ο(n log n) 演算法更快。所以它是一个被广泛使用的算法。在一次采访中，霍尔谈到了发明这个算法的背景。

[霍尔](http://www.nowamagic.net/librarys/veda/tag/%E9%9C%8D%E5%B0%94)出生于斯里兰卡。1956年毕业于牛津大学。然后的两年里他在英国皇家海军服役，他的任务是研究俄国的现代军事，并因此开始学习了俄语。结束服役后， 他作为研究生进入莫斯科大学，主攻计算机翻译。在莫斯科学习了一年。在这个时候正好有一家生产小型科学计算机的公司Elliott Brothers在那里举办展览，霍尔为他们做翻译。当他回国后，这家公司立即聘用了他，连面试都免了。他们还增加了他的工资，因为他会俄语。 Elliott Brothers让霍尔设计一个新的计算机语言。但当他偶尔看到了一篇艾伦·佩利的“算法语言Algol 60报告”(Report on the Algorithmic Language Algol 60)后，他立即推荐公司放弃设计一个新的语言而转为实施ALGOL。公司采纳了他的建议。ALGOL是第一个清晰定义的语言，其语法是用严格公式化的方 法说明的。ALGOL语言并没有被广泛的使用，但它是许多现代程序语言的概念基础。对他个人来说，这个项目不仅为他的事业奠定了基础，还为他带来了甜蜜的婚姻：他跟同组的同事Jill相识并结婚，成为一段美谈。

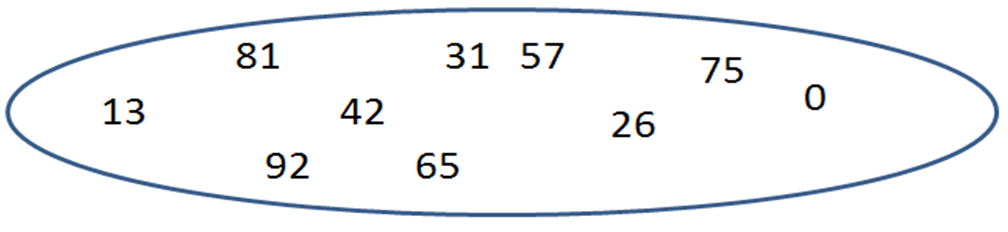
言归正传。1960年代，英国国家物理实验室 (National Physical Laboratory) 开始了一项新的计划：将俄文自动翻译成英文。正好霍尔有这个经历，他与俄国的机器翻译专家相识，还在“机器翻译”(Machine Translation) 上发表过论文。于是他在那里得到了一份工作。

在那个年代，俄文到英文的词汇列表是以字母顺序存储在一条长长的磁带上的。因此，当有一段俄文句子需要翻译时，第一步是把这个句子的词按照同样的顺序排列。这样机器就可以在磁带上只走一遍就可以找到所有的翻译。霍尔意识到，他必须找出一种能在计算机上实现的排序的算法来。他想到的第一个算法是后人称作“冒泡排序 (bubble sort)”的算法。虽然他没有声明这个算法是他发明的，但他显然是独自得到这个算法的。他很快放弃了这个算法，因为它的速度比较慢。用计算复杂度理论(Computational complexity theory) 来说，它平均需要 O(*n*2) 次运算。快速排序 (Quicksort) 是霍尔想到的第二个算法。这个算法的计算复杂度是 O(*n*log*n*) 次运算。当 *n* 特别大的时候，显然步骤要少很多。这个算法是二十世纪七大算法之一，而他本人则被称为影响算法世界的十位大师之一。霍尔自己则认为这个算法是他一生来得到的唯一一个有意义的算法。显然他是谦虚了。太谦虚了！他在计算机语言和数理逻辑上建树颇多。比如，黄富强老师介绍过霍尔逻辑 (Hoare logic)。当时就说我也要写一篇介绍霍尔的文章，没想到竟然拖到现在。

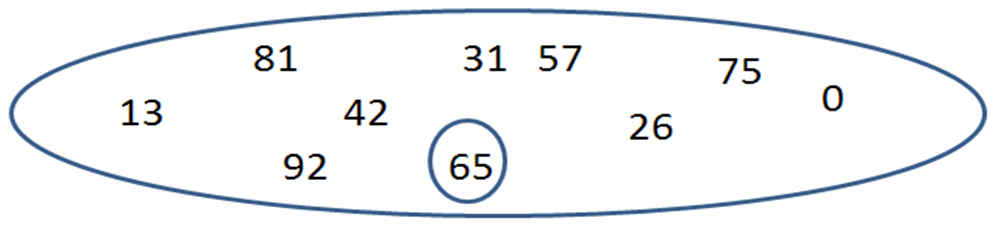
计算机历史博物馆真是一个好去处。还记得我以前写的“建造一架150年前设计的差分机”吗？也是在那里看到的。北京师范大学数学系的王世强教授和沈复兴教授等对“计算复杂度”有一定的研究。我也是从他们那里第一次接触到“计算复杂度”这个问题的。



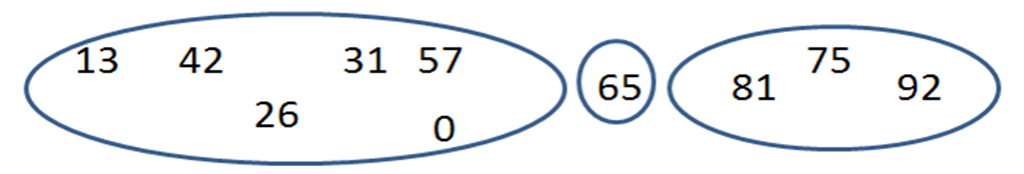
回到“[快速排序](http://www.nowamagic.net/librarys/veda/tag/%E5%BF%AB%E9%80%9F%E6%8E%92%E5%BA%8F)法”，其实“快速排序法”的基本思想是用递归 (Recursion)，每进行一步都将一个大的集合划分为两个小的子集，然后对两个子集实施相同的算法。当两个子集都完成了排序之后再把它们重新粘合到一起。让我们用下面的一个简单例子简单说明一下。



假定我们有一组10个数，我们希望将它们从小到大排列。



我们首先从数列中随机挑出一个元素，称为 "基准"(pivot)。



我们把比这个基准小的数放在左边，把比这个基准大的数放在右边。

上面是我们的第一步。在这一步之后，我们得到了两个小的集合。现在我们重复上面的步骤对这两个小的集合进行排序。这就是我前面说的递归的思想。让我们忽略里面的具体细节。经过一些步骤之后，我们已经将这两个小的集合排好序了。下面的两步就容易理解了。



我把这段故事写出来，希望软件工程师们在学习计算方法时得到一些乐趣。也希望上面的故事和这个小例子能对大家有所帮助。

后注：1991年，杨正瓴老师发现了平均复杂性为 O(nloglogn)的自适应排序算法，以及对独立同分布随机数最坏时间几乎处处为O(n)的排序算法。