## **Quick Sort**

## 快速排序

问题:

将无序序列s进行排序。

## 解法 1:

本问题对无序序列 s 进行升序排序,排序后 s 是从小到大的。

对于长度为 n,下标在左开右闭区间[0,n)中的序列 s,将其分为 left、right 和 mid 三个部分,其中 mid 是中间值,left 和 right 都是无序,left 所有元素小于等于 mid,right 所有元素大于等于 mid。

在每一轮排序中,选取s[0]作为这个 mid 值,将剩下的s[1,n-1]分成 left 和 right 两个部分,分别放在 mid 左右。然后对于 left 和 right,递归的重复上面这个操作,直到 left 和 right 只剩小于等于 3 个元素,则直接进行排序,即可得到升序序列 s,算法结束。

只看第 1 轮,将p = s[0]作为 mid 值,再设置low = 0,high = n - 1。因为 p 已经记录了s[0]的值,我们可以将一个小于等于 p 的值放在这里。当low < high时,high 向左寻找第一个s[high] < p值,该s[high]属于 left 部分,令s[low] = s[high];然后 low 向右寻找第一个s[low] > p,该s[low]属于 right 部分,令s[high] = s[low]。重复上面这个过程,直到 $low \ge high$ ,这时s[0,low-1]都小于等于 p,s[high,n-1]都大于等于 p,令s[low] = p即完成了第 1 轮的放置过程。

对于下面这个长度为 6 的无序序列 s 进行升序排序,初始时p = s[0] = 45,low = 0,high = 5:

0	1	2	3	4	5	
45	90	3	83	17	25	
low					high	

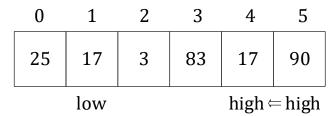
(1) high 向左找到第一个满足s[high] < p即high = 5,令s[low] = s[high],如下图;

0	1	2	3	4	5
25	90	3	83	17	25
low			1	1	high

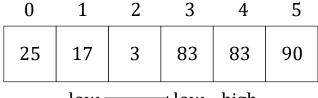
(2) 然后 low 向右找到第一个满足s[low] > p即low = 1, 令s[high] = s[low], 如下图;

0	1	2	3	4	5
25	90	3	83	17	90
low=	high				

(3) 重复第(1)步, high 向左找到第一个满足s[high] < p即high = 4, 令s[low] = s[high], 如下图;

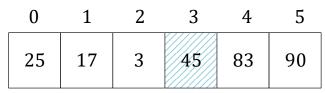


(4) 重复第(2)步,low 向右找到第一个满足s[low] > p即low = 3,令s[high] = s[low],如下图;



low = low high

(5) 重复第(1)步,high 向左找到第一个满足s[high] < p即high = 2,但此时 $low \ge high$ ,直接令s[low] = p,第 1 轮的放置结束,分别对s[0,2]和s[4,5]继续这个过程,直到 left 和 right 的长度小于等于 1,最终整个序列都是升序的,算法结束,如下图;



high <del>← low</del> high

对于长度 n 的序列 s,每一轮放置所需要的时间为O(n),总共需要 $\log_2 n$ 轮,该算法的时间复杂度为 $O(n \cdot \log_2 n)$ 。