

# Insert Sort

## 插入排序

问题:

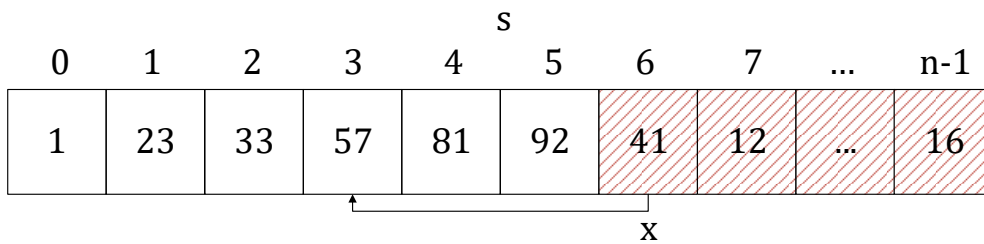
将长度为 $n$ 的无序序列 $s$ 进行排序。

解法:

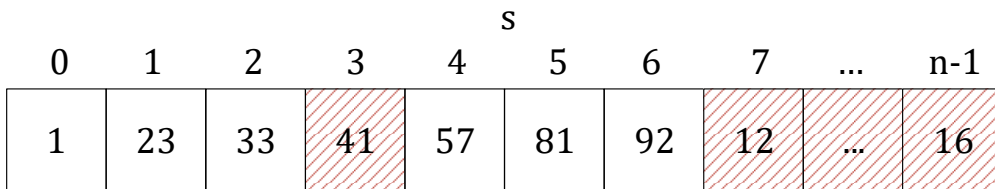
本问题对无序序列 $s$ 进行升序排序，排序后 $s$ 是从小到大的。

将长度为 $n$ 的序列 $s$ 分为左右两个部分 $left$ 和 $right$ ， $left$ 是已序的，范围为 $s[0, k]$ ， $right$ 是未序的，范围为 $s[k + 1, n - 1]$ ，其中 $0 \leq k < n$ 。对 $right$ 中最左边的元素 $x = s[k + 1]$ ，在 $left$ 部分中找到一个位置 $i$ ，满足 $s[i - 1] \leq x \leq s[i]$ ，也就是说 $x$ 可以夹在 $s[i - 1]$ 和 $s[i]$ 之间。为了满足 $left$ 的有序性，将 $left$ 中 $s[i, k]$ 部分的元素向右移动一个位置到 $s[i + 1, k + 1]$ ，将 $x$ 放置在原 $s[i]$ 的位置即可。

例如下图中， $left$ 部分为 $s[0, 5]$ ， $right$ 部分为 $s[6, n - 1]$ ， $right$ 最左边的首部元素 $x = s[6] = 41$ ，在 $left$ 部分中合适的插入位置为 $i = 3$  ( $s[2] \leq x \leq s[3]$ )。



将 $s[3, 5]$ 向右移动一位到 $s[4, 6]$ ，将原 $x$ 移动到 $s[3]$ ，就完成了一次插入。



对于长度为 $n$ 的数组 $s$ ，将 $left$ 初始化为 $s[0, 0]$ ， $right$ 初始化为 $s[1, n - 1]$ 。重复上面的插入操作，直到 $right$ 为空，这时 $left$ 部分即为已序的结果，算法结束。对长度为 $n$ 的序列 $s$ ，每一轮将 $right$ 中一个元素插入 $left$ 中的时间为 $O(n)$ ，总共需要 $n$ 轮操作，该算法的时间复杂度为 $O(n^2)$ 。