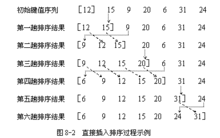
插入排序是一种简单直观且稳定的[排序算法](https://baike.baidu.com/item/%E6%8E%92%E5%BA%8F%E7%AE%97%E6%B3%95/5399605)。如果有一个已经有序的数据序列，要求在这个已经排好的数据序列中插入一个数，但要求插入后此数据序列仍然有序，这个时候就要用到一种新的排序方法——[插入排序法](https://baike.baidu.com/item/%E6%8F%92%E5%85%A5%E6%8E%92%E5%BA%8F%E6%B3%95/612261),插入排序的基本操作就是将一个数据插入到已经排好序的有序数据中，从而得到一个新的、个数加一的有序数据，算法适用于少量数据的排序，[时间复杂度](https://baike.baidu.com/item/%E6%97%B6%E9%97%B4%E5%A4%8D%E6%9D%82%E5%BA%A6/1894057" \t "_blank)为O(n^2)。是稳定的排序方法。插入算法把要排序的[数组](https://baike.baidu.com/item/%E6%95%B0%E7%BB%84/3794097" \t "_blank)分成两部分：第一部分包含了这个数组的所有元素，但将最后一个元素除外（让数组多一个空间才有插入的位置），而第二部分就只包含这一个元素（即待插入元素）。在第一部分排序完成后，再将这个最后元素插入到已排好序的第一部分中。

插入排序的基本思想是：每步将一个待排序的记录，按其关键码值的大小插入前面已经排序的文件中适当位置上，直到全部插入完为止。

正如生活中整理扑克的方法：一张一张的来，将每一张扑克插入到其他已经有序的扑克中的适当位置。

将n个元素的数列分为已有序和无序两个部分，如

[](https://baike.baidu.com/pic/%E6%8F%92%E5%85%A5%E6%8E%92%E5%BA%8F/7214992/0/d57e99942da24e5dd21b7080?fr=lemma&ct=single)插入排序过程示例

下所示：

{{a1}，{a2，a3，a4，…，an}}

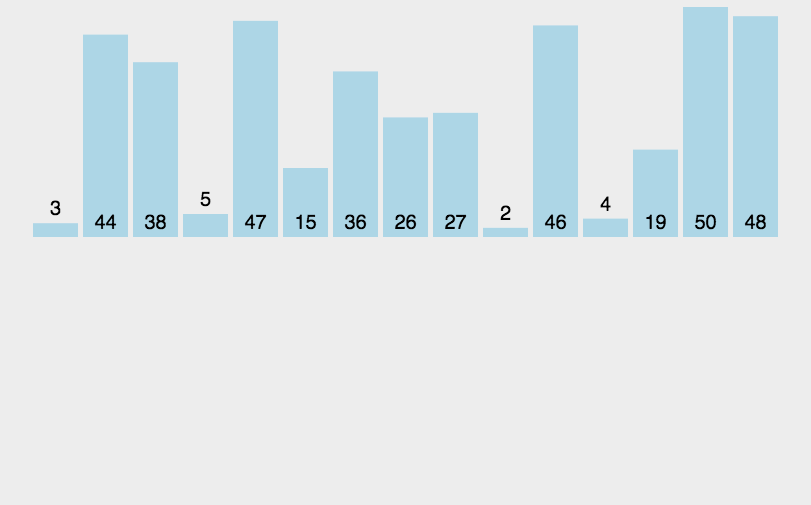
{{a1⑴，a2⑴}，{a3⑴，a4⑴ …，an⑴}}

…

{{a1(n-1），a2(n-1) ，…},{an(n-1)}}

每次处理就是将无序数列的第一个元素与有序数列的元素从后往前逐个进行比较，找出插入位置，将该元素插入到有序数列的合适位置中。

假设在一个无序的数组中，要将该数组中的数按插入排序的方法从小到大排序。假设啊a[]={3,5,2,1,4};插入排序的思想就是比大小，满足条件交换位置，一开始会像冒泡排序一样，但会比冒泡多一步就是交换后（a[i]=a[i+1]后）原位置（a[i]）会继续和前面的数比较满足条件交换，直到a[i+1]前面的数组是有序的。



一般来说，插入排序都采用in-place在[数组](https://baike.baidu.com/item/%E6%95%B0%E7%BB%84)上实现。具体算法描述如下：

⒈ 从第一个元素开始，该元素可以认为已经被排序

⒉ 取出下一个元素，在已经排序的元素序列中从后向前扫描

⒊ 如果该元素（已排序）大于新元素，将该元素移到下一位置

⒋ 重复步骤3，直到找到已排序的元素小于或者等于新元素的位置

⒌ 将新元素插入到下一位置中

⒍ 重复步骤2~5

如果比较操作的代价比交换操作大的话，可以采用[二分查找](https://baike.baidu.com/item/%E4%BA%8C%E5%88%86%E6%9F%A5%E6%89%BE)法来减少比较操作的数目。该算法可以认为是插入排序的一个变种，称为[二分查找](https://baike.baidu.com/item/%E4%BA%8C%E5%88%86%E6%9F%A5%E6%89%BE)排序。