

7 月 10 日 - 7 月 11 日	常用排序 + 堆 + 优先队列 + 二分	林凯
---------------------	----------------------	----

基础培训

1.常用排序

基础篇:

冒泡排序、插入排序、希尔排序、选择排序

时间复杂度: $O(N) - O(N^2)$

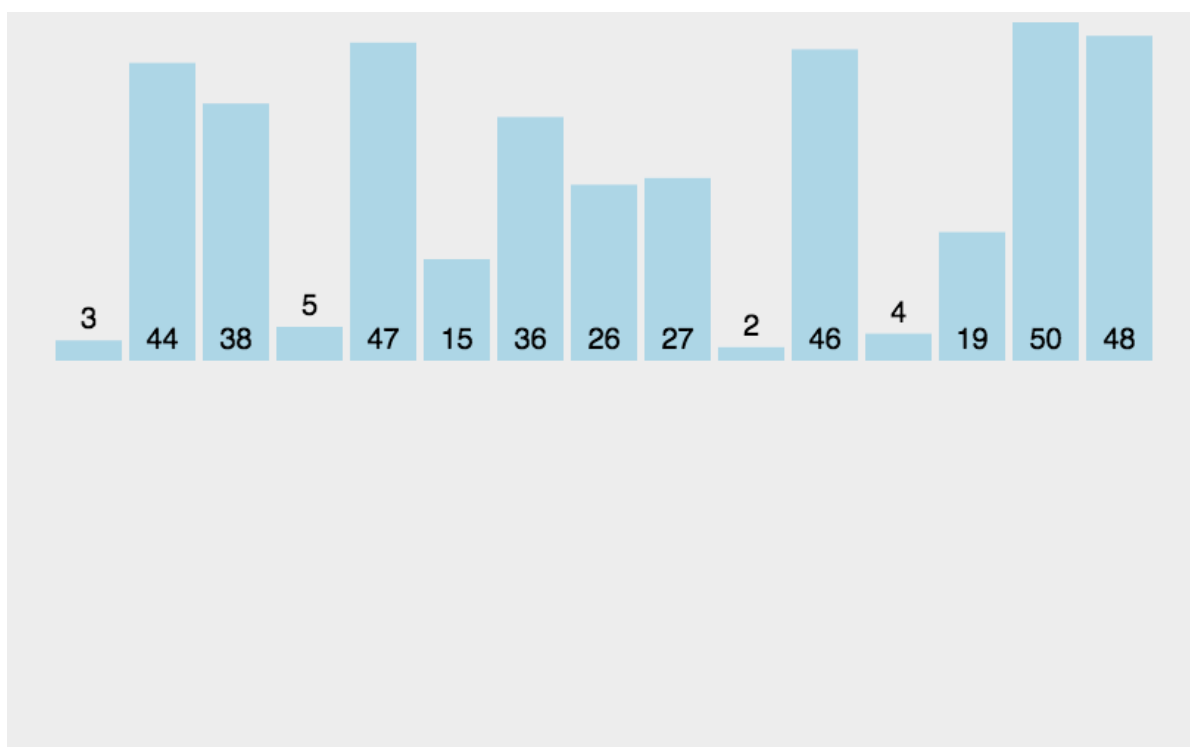
冒泡排序

直接插入排序

对于数组中的每个数,找到他目前合适的位置

特点: 元素集合越**接近有序**, 直接插入排序算法的时间效率越高

稳定性: 稳定



希尔排序

选择排序的优化版

稳定性：不稳定，相等的关键字可能会被划分到不同的组进行预排序

运用缩小增量法：

1.间隔分组

2.组内排序

3.重新设置分组

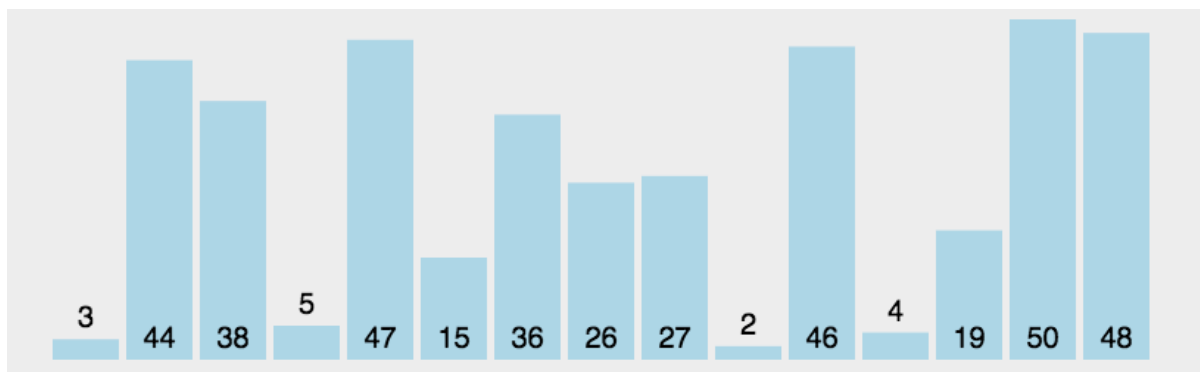
4.插入排序

选择排序 $O(n^2)$

稳定性：不稳定

用的少

从小到大一个个找



进阶篇：

快速排序,归并排序

快速排序

分治思想

快排属于分治算法，分治算法都有三步：

分成子问题

递归处理子问题

子问题合并

稳定性:不稳定

1.确定分界点

2.调整区间

3.递归处理左右两段

归并排序

稳定性:稳定

堆

- 1.插入一个数
- 2.求集合当中的最小值
- 3.删除最小值
- 4.删除任意一个元素
- 5.修改任意一个元素

优先队列

和队列基本操作相同:

- top 访问队头元素
- empty 队列是否为空
- size 返回队列内元素个数
- push 插入元素到队尾 (并排序)
- emplace 原地构造一个元素并插入队列
- pop 弹出队头元素
- swap 交换内容

二分

手写二分

二分答案

stl二分

二分查找

`lower_bound()`和`upper_bound()`，`binary_search()`都是利用二分查找的方法在一个排好序的数组中进行查找的。

`lower_bound()`：返回大于或等于目标值的第一个位置

`upper_bound()`：返回大于目标值的第一个位置

`binary_search()`：若目标值存在则返回true，否则返回false