**排序算法：**

****

**插入排序：**

将第一个元素视为一个有序序列，对于未排序数据，在已排序序列中从后向前扫描，找到相应位置并插入。

**希尔排序（插入排序改进版）：**

1.插入排序在对几乎已排好序的数据进行操作时，效率很高。

2.插入排序相对较低效，一次只能将数据移动一位。

希尔排序通过将全部元素分为几个区域来提升插入排序的性能。这样可以让一个元素一次性朝最终位置前进一大步。然后取越来越小的步长进行排序，最后一步就是普通的插入排序。（步长的选择为一般为n/2，性能更加优越的步长选择…………）

**选择排序：**

每一轮选出最小（大）的元素，放到已排序序列的末尾（初始位置）。

**堆排序：**

堆的定义：

大顶堆：每个节点的值都大于或等于其子节点；小顶堆：每个节点的值都小于或等于其子节点。

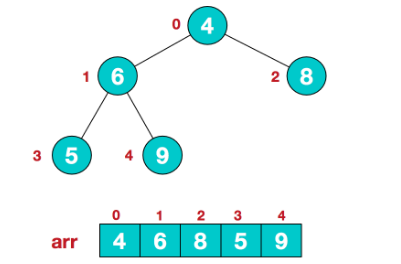
堆的构建：

从最后一个非叶子节点开始，调整节点及其子节点，使其满足堆的定义。

1.将无序序列构建成堆，根据需求选择大顶堆（升序）或小顶堆（降序）。

2.将堆顶元素与末尾元素交换，将最大元素放到末端。

3.将剩余未排序序列重新调整，使其满足堆定义，反复执行2、3，直到整个序列有序。



**冒泡排序：**

比较相邻的元素，如果第一个比第二个大就交换。

**快速排序：**

1.从序列中挑选出一个基准值（pivot）。

2.重新排列序列，将比基准值小的放在基准前面，将比基准值大的放在基准后面。

3.递归地将小于基准值的子序列和大于基准值的子序列重新排列，直到序列的长度为0或1。

**归并排序：**

采用分治法：1. 分割：递归地把当前序列平均分割成两半。 2. 集成：在保持元素顺序的同时将上一步得到的子序列集成到一起（归并）。

