三种排序算法的时间复杂度都是 O(nlogn)

一般情况下，就运行时间而言：

快速排序<归并排序<堆排序

三种排序算法的缺点：

快速排序：极端情况下排序效率低

归并排序：需要额外的内存开销（在归并时要开一个临时列表）

堆排序：在快的排序算法中相对较慢

排序方法	时间复杂度			空间复杂度	稳定性	代码复杂度
	最坏情况	平均情况	最好情况			
冒泡排序	$O(n^2)$	$O(n^2)$	O(n)	O(1)	稳定	简单
直接选择排序	$O(n^2)$	$O(n^2)$	$O(n^2)$	O(1)	不稳定	简单
直接插入排序	$O(n^2)$	$O(n^2)$	$O(n^2)$	O(1)	稳定	简单
快速排序	$O(n^2)$	O(nlogn)	O(nlogn)	平均情况O(logn)；最坏情况O(n)	不稳定	较复杂
堆排序	O(nlogn)	O(nlogn)	O(nlogn)	O(1)	不稳定	复杂
归并排序	O(nlogn)	O(nlogn)	O(nlogn)	O(n)	稳定	较复杂

稳定性：当列表中有相同的元素值时，能保证他们的相对位置不变。

（如果是挨着换的那种就稳定，如果是飞着换的就不稳定）

例如：1，6，4，5，6，8，6

Python 的 sort 函数使用的是 Timsort 排序算法。Timsort 算法是一种融合了合并排序和插入排序的稳定排序算法，它在数据集中已经有序或部分有序的情况下，具有很快的速度和很小的空间消耗，而且也适合处理大大小小的数据集。Timsort 算法是 Python 中默认的排序算法，被用于内置的列表排序函数（sort）和非稳定排序函数（sorted）。