1.对一组数据进行排序

1）数据特性？

A.

含有大量重复数据：三路快排（？）

不含大量重复数据：普通快排（快速排序的多种实现）

B.

是否近乎有序（如银行业务）：插入排序

C.

数据范围有限（如学生成绩）：计数排序

D.

排序有何额外需求，如是否需要稳定排序：归并排序

E.

数据具体的存储状况（数据背后的数据结构），如采用链表存储：归并排序

数据具体的存储状况,如数据量与内存大小不匹配：外排序（？）

F.

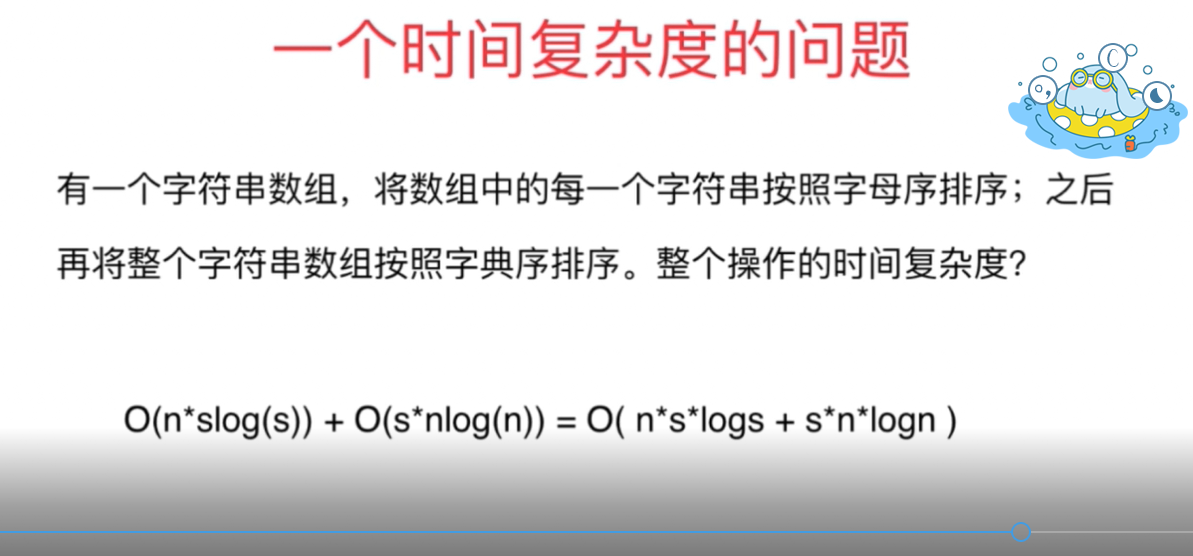
其他情况

2.时间复杂度

n表示数据规模，O(f(n))表示运行算法需要执行的指令数，与f(n)成正比

一般O指算法执行的最低上界

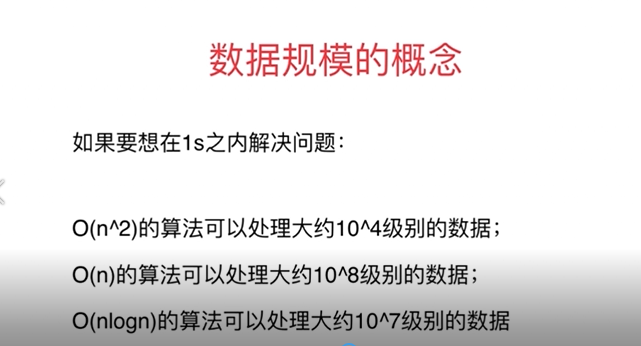
O(nlogn+n)= O(nlogn)，O(nlogn+n^2)= O(n^2)，n指向相同



其中s指字符串长度，n指数组内字符串个数

3.数据规模

时间复杂度



保守起见，一般会把数据量除以10（或者宽松点除以2）

空间复杂度

A.

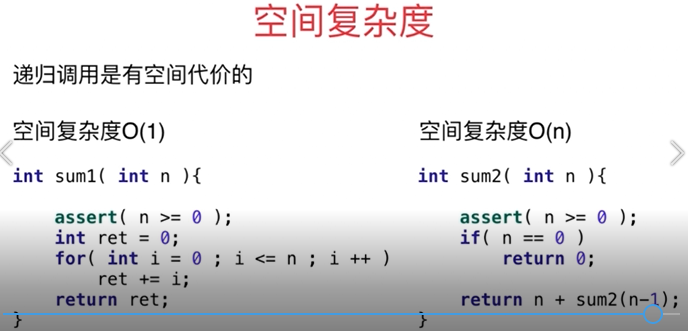
多开常数空间O(1)

多开一个辅助数组O(n)

多开一个辅助二维数组O(n^2)

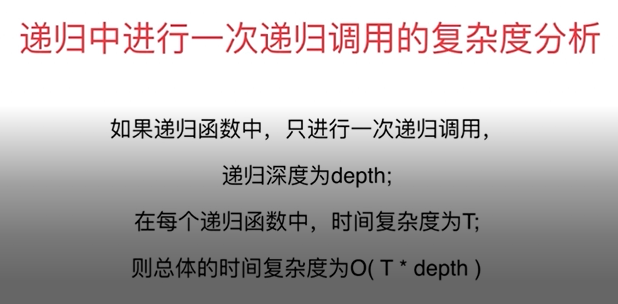
B.

递归有空间代价，递归深度=空间复杂度



4.递归算法复杂度

每一次递归中进行一次递归调用



每一次递归中进行多次递归调用（视频中有两棵递归树，一棵深度为n，所以算法复杂度O(n^2)，另一颗深度logn，所以算法复杂度O(nlogn)，可以看下下面的链接里面有推导）

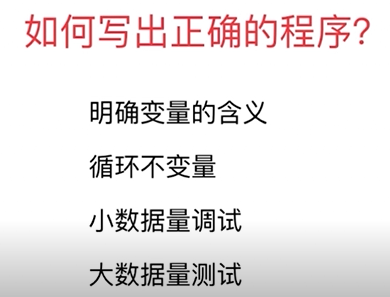
[算法导论------递归算法的时间复杂度求解](https://blog.csdn.net/so_geili/article/details/53444816)

5.均摊复杂度

复杂度均摊（复杂度震荡）

可以创建自己的动态数组、动态栈、动态队列

6.如何写出正确程序



7.数组问题

题目（空间复杂度O(1)）

283[移动零](https://leetcode-cn.com/problems/move-zeroes/)

27[移除元素](https://leetcode-cn.com/problems/remove-element/)

26[删除排序数组中的重复项](https://leetcode-cn.com/problems/remove-duplicates-from-sorted-array/)没想出更好的方法

80[删除排序数组中的重复项 II](https://leetcode-cn.com/problems/remove-duplicates-from-sorted-array-ii/)

快速排序

<https://www.cnblogs.com/yangecnu/p/Introduce-Quick-Sort.html>

这篇比较全，概念清晰，代码没有好好看过

<https://blog.csdn.net/puppylpg/article/details/70233728>

这篇是简写

<https://blog.csdn.net/k_koris/article/details/80585979>

双路快排和三路快排

<https://segmentfault.com/a/1190000014103659?utm_source=tag-newest>

双路快排用到的双指针法及其具体应用