jing<https://github.com/itdevbooks/pdf> it方面的书

## √第1节 认识复杂度、对数器、二分法

### 选择排序35:05

### 冒泡排序37:55

相邻交换

### 插入排序44:49 52:59代码

向左看

### 在一个有序数组中，找某个数是否存在 1:50:12 2:01:22代码

### 在一个有序数组中，找>=某个数最左侧的位置 1:57:16

### 在一个有序数组中，找<=某个数最右侧的位置

### 局部最小值问题2:12:13

## √第2节 异或运算相关面试题

### 题目一：如何不用额外变量交换两个数 17：00

### 题目二 ：一个数组中有一种数出现了奇数次，其他数都出现了偶数次，怎么找到并打印这种数 30:45

### 题目三 ：怎么把一个int类型的数，提取出最右侧的1来 36:33

### 题目四 ：一个数组中有两种数出现了奇数次，其他数都出现了偶数次，怎么找到并打印这两种数 43:11

### 题目5：KM 1:06:12 1:25:49代码

一个数组中有一种数出现K次，其他数都出现了M次，

M > 1, K < M

找到，出现了K次的数，

要求，额外空间复杂度O(1)，时间复杂度O(N)

## √第3节 一些基础的数据结构

### 单链表和双链表如何反转 13:45单 24:21双

### 把给定值都删除 29:20

### 栈和队列的实际实现36:19 48:41代码

双向链表实现

数组实现

怎么用数组实现不超过固定大小的队列和栈？

栈：正常使用

队列：环形数组

### 实现一个特殊的栈，在基本功能的基础上，再实现返回栈中最小元素的功能 1:03:38

1）pop、push、getMin操作的时间复杂度都是 O(1)。

2）设计的栈类型可以使用现成的栈结构。

### 如何用栈结构实现队列结构 1:11:17 1:20:24代码

如何用队列结构实现栈结构 1:23:27

### 求数组arr[L..R]中的最大值，怎么用递归方法实现。 1:26:46

1）将[L..R]范围分成左右两半。左：[L..Mid] 右[Mid+1..R]

2）左部分求最大值，右部分求最大值

3） [L..R]范围上的最大值，是max{左部分最大值，右部分最大值}

注意：2）是个递归过程，当范围上只有一个数，就可以不用再递归了

### Master公式1:51:56

形如

T(N) = a \* T(N/b) + O(N^d)(其中的a、b、d都是常数)

的递归函数，可以直接通过Master公式来确定时间复杂度

如果 log(b,a) < d，复杂度为O(N^d)

如果 log(b,a) > d，复杂度为O(N^log(b,a))

如果 log(b,a) == d，复杂度为O(N^d \* logN)

### 哈希表和有序表2:13:43

红黑树、AVL树、size-balance-tree和跳表等都属于有序表结构，只是底层具体实现不同

## √第4节 归并排序及其相关面试题

### √归并排序递归04:15相等的时候，拷贝左边的

二分递归，然后合并

### √归并排序迭代28:00

### √√小和52:40 相等的时候，先拷贝右边的

在一个数组中，一个数左边比它小的数的总和，叫数的小和，所有数的小和累加起来，叫数组小和。求数组小和。

例子： [1,3,4,2,5]

1左边比1小的数：没有

3左边比3小的数：1

4左边比4小的数：1、3

2左边比2小的数：1

5左边比5小的数：1、3、4、 2

所以数组的小和为1+1+3+1+1+3+4+2=16

### √√逆序对1:27:50从右往左，相等拷贝右

在一个数组中，

任何一个前面的数a，和任何一个后面的数b，

如果(a,b)是降序的，就称为逆序对

返回数组中所有的逆序对

### √√小于2倍数 1:35:22

在一个数组中，

对于每个数num，求有多少个后面的数 \* 2 依然<num，求总个数

比如：[3,1,7,0,2]

3的后面有：1，0

1的后面有：0

7的后面有：0，2

0的后面没有

2的后面没有

所以总共有5个

1.条件改成2倍，测试不通过

2.窗口

## 第5节 归并排序附加题、随机快速排序

### 半√归并排序补充题目（难）4:00 力扣327

题目描述：

https://leetcode.com/problems/count-of-range-sum/

给定一个数组arr，两个整数lower和upper，

返回arr中有多少个子数组的累加和在[lower,upper]范围上

前缀和

i位置结尾

1:21:52大体步骤

归并排序。根据右数组求出范围，然后遍历左数组两个指针，双指针之差。<=max，<min

### √荷兰国旗问题 1:28:13 1:37:47加强版

给定一个数组arr，和一个整数num。请把小于num的数放在数组的左边，等于num的数放在中间，大于num的数放在数组的右边。

要求额外空间复杂度O(1)，时间复杂度O(N)

### √快排1:46:29

2:06:00

快排2.02:08:13

2:36:42非递归版本

## 第6节 堆和堆排序

### 比较器05:00

### 堆 39:51

heapinsert 1:06:09往上看，上移

删除最值，下沉

1:38:48比较

### √堆排序2:20:37下沉

从右往左，循环下沉，遍历一次。然后慢慢找最大值，放数组最后。

### √已知一个几乎有序的数组。几乎有序是指，如果把数组排好顺序的话，每个元素移动的距离一定不超过k，并且k相对于数组长度来说是比较小的。 2:30:22

请选择一个合适的排序策略，对这个数组进行排序。

## 第7节 加强堆

### √√最大线段重合问题（用堆的实现）16:24 55:20代码

给定很多线段，每个线段都有两个数[start, end]，

表示线段开始位置和结束位置，左右都是闭区间

规定：

1）线段的开始和结束位置一定都是整数值

2）线段重合区域的长度必须>=1

返回线段最多重合区域中，包含了几条线段

找到最小开始和最大结束

根据开始位置排序[1,7], [1,5], [1,6]

### 手动改写堆题目练习1:05:13 1:49:04开始

给定一个整型数组，int[] arr；和一个布尔类型数组，boolean[] op

两个数组一定等长，假设长度为N，arr[i]表示客户编号，op[i]表示客户操作

arr = [ 3 , 3 , 1 , 2, 1, 2, 5…

op = [ T , T, T, T, F, T, F…

依次表示：3用户购买了一件商品，3用户购买了一件商品，1用户购买了一件商品，2用户购买了一件商品，1用户退货了一件商品，2用户购买了一件商品，5用户退货了一件商品…

### 1:15:51加强堆

反向索引表

## 第8节 前缀树、不基于比较的排序、排序稳定性

### 前缀树（prefix tree trie）16:44

设计一种结构。用户可以：

1）void insert(String str) 添加某个字符串，可以重复添加，每次算1个

2）int search(String str) 查询某个字符串在结构中还有几个

3) void delete(String str) 删掉某个字符串，可以重复删除，每次算1个

4）int prefixNumber(String str) 查询有多少个字符串，是以str做前缀的

### 计数排序1:09:54

年龄【0,200】

### 基数排序1:16:09

非负整数，从个位到高位

1:38:45

## 第9节 排序总结、链表相关面试题

### √在整型数组中，请把奇数放在数组左边，偶数放在数组右边，要求所有奇数之间原始的相对次序不变，所有偶数之间原始相对次序不变。 21:13

做不到，如果能做到，快排早就稳定了。

对数据做了严格的限定，不用深入研究了。

### Array.sort 30:00

基础类型用快排

引用类型用归并

数据量小的时候，有插入排序

### √快慢指针43:19

1）输入链表头节点，奇数长度返回中点，偶数长度返回上中点

2）输入链表头节点，奇数长度返回中点，偶数长度返回下中点

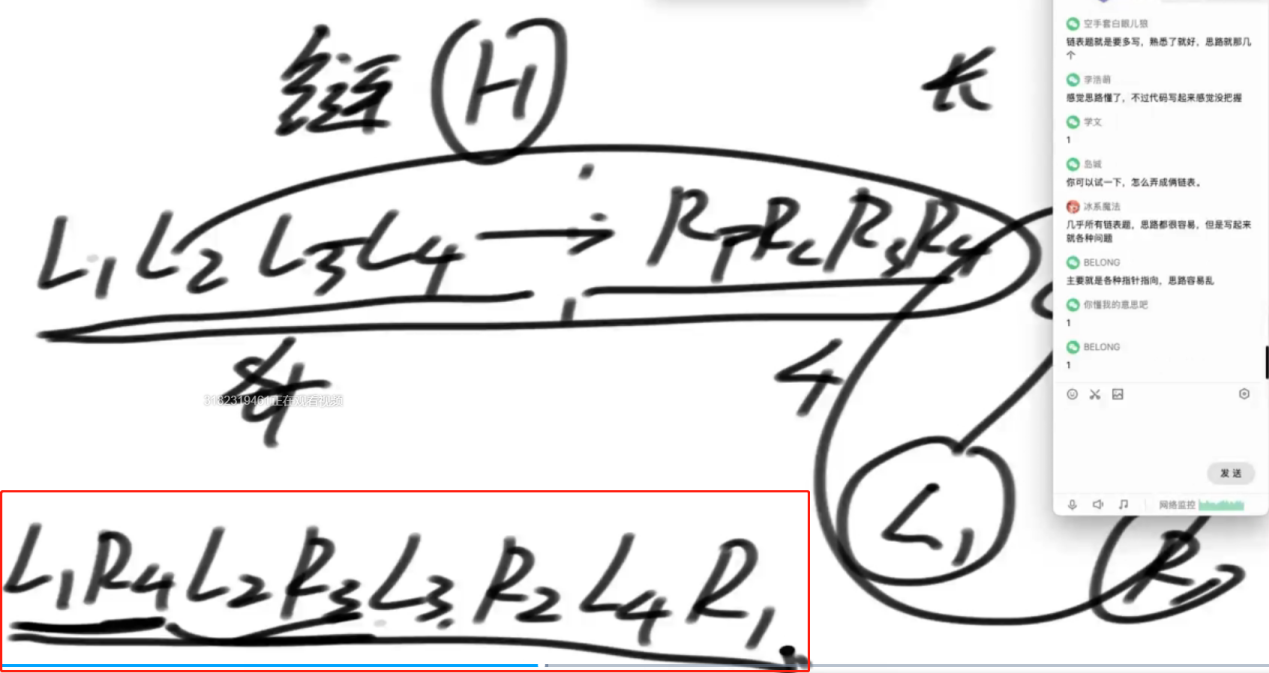
3）输入链表头节点，奇数长度返回中点前一个，偶数长度返回上中点前一个

4）输入链表头节点，奇数长度返回中点前一个，偶数长度返回下中点前一个

### √√给定一个单链表的头节点head，请判断该链表是否为回文结构。 51:28

1）哈希表方法特别简单（笔试用）

2）改原链表的方法就需要注意边界了（面试用）



### √将单向链表按某值划分成左边小、中间相等、右边大的形式 1:09:05

1）把链表放入数组里，在数组上做partition（笔试用）

2）分成小、中、大三部分，再把各个部分之间串起来（面试用）

### √√有随机节点的链表1:42:42

一种特殊的单链表节点类描述如下

class Node {

int value;

Node next;

Node rand;

Node(int val) { value = val; }

}

rand指针是单链表节点结构中新增的指针，rand可能指向链表中的任意一个节点，也可能指向null。

给定一个由Node节点类型组成的无环单链表的头节点 head，请实现一个函数完成这个链表的复制，并返回复制的新链表的头节点。

【要求】

时间复杂度O(N)，额外空间复杂度O(1)

1.用map

2.不用容器

老新

### 两个链表相交2:06:18未讲

给定两个可能有环也可能无环的单链表，头节点head1和head2。请实现一个函数，如果两个链表相交，请返回相交的 第一个节点。如果不相交，返回null

【要求】

如果两个链表长度之和为N，时间复杂度请达到O(N)，额外空间复杂度 请达到O(1)。

## √第10节 二叉树基本算法（上）

### √√给定两个可能有环也可能无环的单链表，头节点head1和head2。请实现一个函数，如果两个链表相交，请返回相交的 第一个节点。如果不相交，返回null 07:34

【要求】

如果两个链表长度之和为N，时间复杂度请达到O(N)，额外空间复杂度 请达到O(1)。

1.hashset

2.快慢指针

### √√给定两个可能有环也可能无环的单链表，头节点head1和head2。请实现一个函数，如果两个链表相交，请返回相交的 第一个节点。如果不相交，返回null 32:39

【要求】

如果两个链表长度之和为N，时间复杂度请达到O(N)，额外空间复杂度 请达到O(1)。

### 二叉树的先序、中序、后序遍历1:07:30

### X祖先节点交集，A交B 1:21:08

### 二叉树的先序、中序、后序遍历 非递归1:36:27

根右左

### 怎么用一个栈实现后续遍历？1:47:00

### 中序遍历1:49:44

弹出打印

/

## 第11节 二叉树基本算法（下）

### 能不能不给单链表的头节点，只给想要删除的节点，就能做到在链表上把这个点删掉？05:00

### 实现二叉树的按层遍历13:43

### 二叉树的序列化和反序列化20:19

反序列化36:21

中序没法序列化

按层序列化40:54

### 付费431.将N叉树编码为二叉树1:01:44

### 如何设计一个打印整棵树的打印函数 1:21:31很少考了

### 求二叉树最宽的层有多少个节点 1:25:23

当后

下后

### 返回该节点的后继节点1:45:09

二叉树结构如下定义：

Class Node {

V value;

Node left;

Node right;

Node parent;

}

给你二叉树中的某个节点，返回该节点的后继节点

### 纸的折痕2:07:57

请把一段纸条竖着放在桌子上，然后从纸条的下边向上方对折1次，压出折痕后展开。此时折痕是凹下去的，即折痕突起的方向指向纸条的背面。 如果从纸条的下边向上方连续对折2次，压出折痕后展开，此时有三条折痕，从上到下依次是下折痕、下折痕和上折痕。

给定一个输入参数N，代表纸条都从下边向上方连续对折N次。 请从上到下打印所有折痕的方向。

例如:N=1时，打印: down N=2时，打印: down down up

## 第12节 二叉树的基本算法+二叉树的递归套路

### √√判断二叉树是否是完全二叉树 5:55

### √√给定一棵二叉树的头节点head，返回这颗二叉树是不是平衡二叉树 19:53

### √√给定一棵二叉树的头节点head，返回这颗二叉树是不是搜索二叉树 36:29

### 给定一棵二叉树的头节点head，任何两个节点之间都存在距离， 返回整棵二叉树的最大距离 58:35

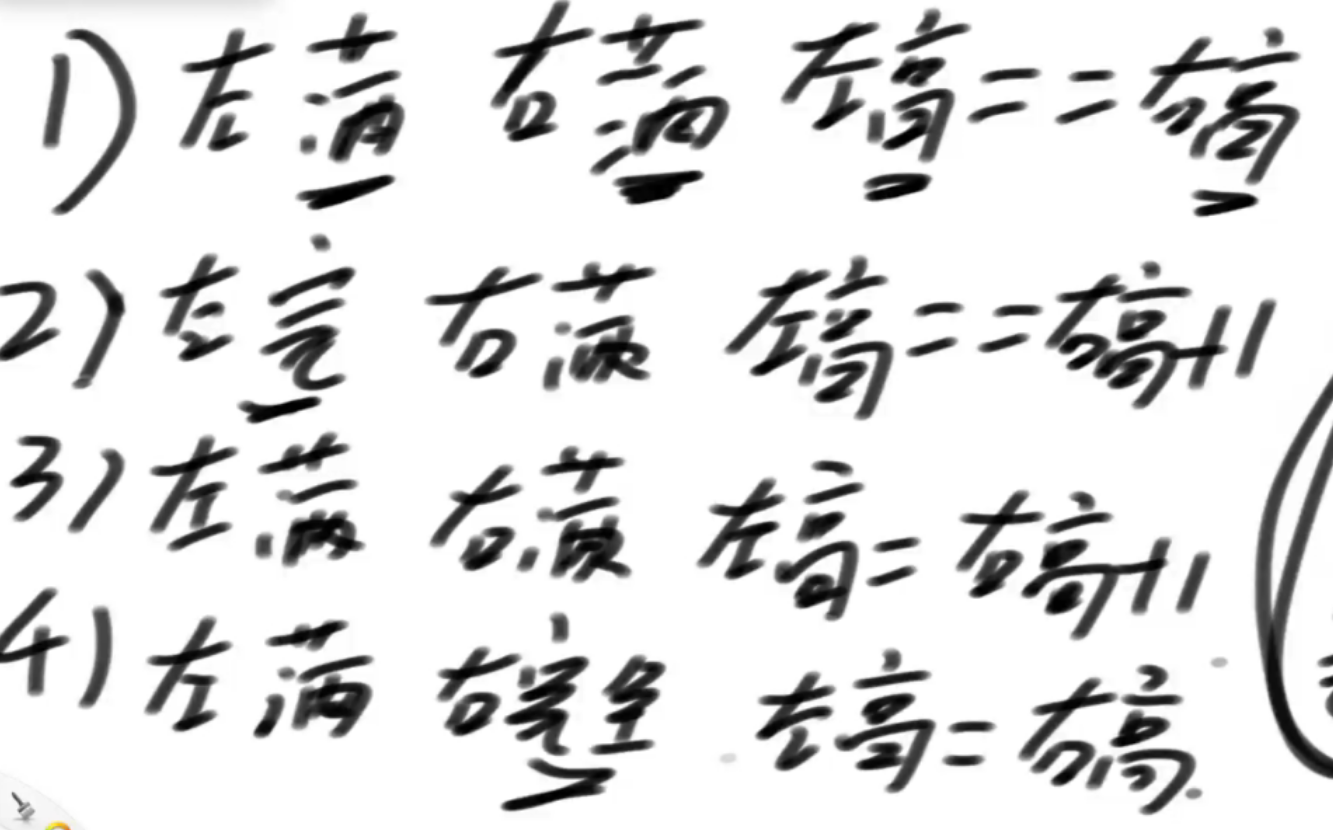
### √√给定一棵二叉树的头节点head，返回这颗二叉树是不是满二叉树 1:15:20

### 给定一棵二叉树的头节点head， 返回这颗二叉树中最大的二叉搜索子树的大小 1:24:00

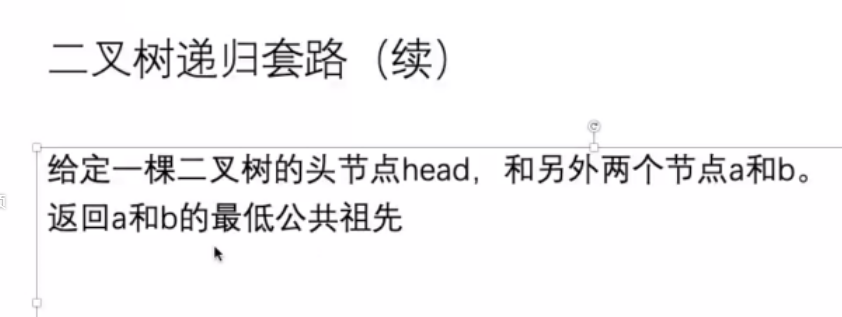
## 第13节 贪心算法（上）

### 判断二叉树是否是完全二叉树？如何用递归套路。6:53

四种情况



### 公共祖先28:43



### 多叉树51:41

派对的最大快乐值 第12节里面的

公司的每个员工都符合 Employee 类的描述。整个公司的人员结构可以看作是一棵标准的、 没有环的多叉树。树的头节点是公司唯一的老板。除老板之外的每个员工都有唯一的直接上级。 叶节点是没有任何下属的基层员工(subordinates列表为空)，除基层员工外，每个员工都有一个或多个直接下级。

### √√给定一个由字符串组成的数组strs， 必须把所有的字符串拼接起来， 返回所有可能的拼接结果中，字典序最小的结果1:10:30

### 小人右走下走01:14:16 贪心反例

## 第14节 贪心算法（下）

### √√会议04:11

一些项目要占用一个会议室宣讲，会议室不能同时容纳两个项目的宣讲。

给你每一个项目开始的时间和结束的时间

你来安排宣讲的日程，要求会议室进行的宣讲的场次最多。

返回最多的宣讲场次。

### √√金条28:33

一块金条切成两半，是需要花费和长度数值一样的铜板的。

比如长度为20的金条，不管怎么切，都要花费20个铜板。 一群人想整分整块金条，怎么分最省铜板?

例如,给定数组{10,20,30}，代表一共三个人，整块金条长度为60，金条要分成10，20，30三个部分。

如果先把长度60的金条分成10和50，花费60; 再把长度50的金条分成20和30，花费50;一共花费110铜板。

但如果先把长度60的金条分成30和30，花费60;再把长度30金条分成10和20， 花费30;一共花费90铜板。

输入一个数组，返回分割的最小代价。

### 项目资金和利润52:27

输入: 正数数组costs、正数数组profits、正数K、正数M

costs[i]表示i号项目的花费

profits[i]表示i号项目在扣除花费之后还能挣到的钱(利润)

K表示你只能串行的最多做k个项目

M表示你初始的资金

说明: 每做完一个项目，马上获得的收益，可以支持你去做下一个项目。不能并行的做项目。

输出：你最后获得的最大钱数。

### √√点灯1:14:56

给定一个字符串str，只由‘X’和‘.’两种字符构成。

‘X’表示墙，不能放灯，也不需要点亮

‘.’表示居民点，可以放灯，需要点亮

如果灯放在i位置，可以让i-1，i和i+1三个位置被点亮

返回如果点亮str中所有需要点亮的位置，至少需要几盏灯

### 并查集1:27:41

## 第15节 并查集及其相关题目

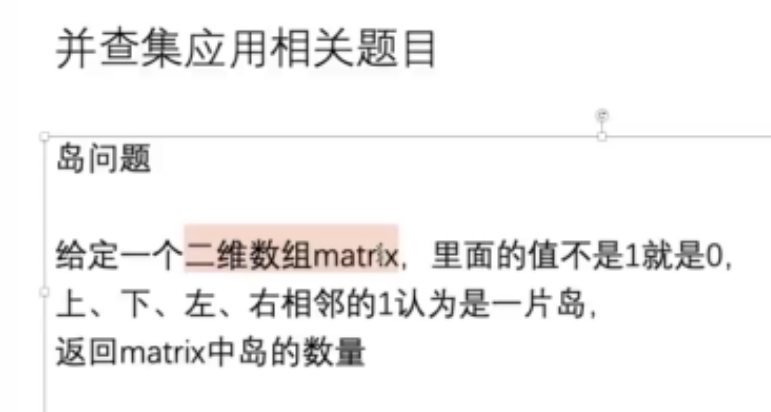
### 547.省份数量13:05

二维数组变成并查集

### √√岛问题 35:43

### 305岛屿数量II 1:32:47

### √√岛问题1:56:29并行



## 第16节 图

### 两种表达方式9:15

1.邻接表。感觉是描述点。

节点1：下一个节点1（可加上权重1），下一个节点2（可加上权重2）。

A：B，C。A的下一个节点是B和C。

2.邻接矩阵。

二维数组。描述的是边。

3.点、边、点 的数组。感觉是描述边。

代码19:00

### 宽度优先遍历37:00

需要一个队列。入队打印和出队打印都可以。

需要一个set，判断是否存在

### 深度优先遍历38:00

需要一个栈。入栈就打印。

需要一个set，判断是否存在

### 拓扑排序50:48

编译顺序

工作流

<https://www.lintcode.com/problem/>

### √√最小生成树算法4kruskal1:29:30对点并查集，贪心

从最小边开始，不构成环的边就是符合要求的。

### √√Prim算法1:39:50对边取最小，贪心

某个点的最小边肯定会被选上

### √√dijkatra算法2:20:20

## 第17节 认识一些经典递归过程

### dijkatra优化3:49

### 暴力递归34:46

### 打印n层汉诺塔从最左边移动到最右边的全部过程 35:25

### 打印一个字符串的全部子序列 55:20

### 打印一个字符串的全部子序列，要求不要出现重复字面值的子序列1:12:18

### 打印一个字符串的全部排列 1:14:12

### 打印一个字符串的全部排列，要求不要出现重复的排列1:40:30

### 逆序的栈1:44:41

给你一个栈，请你逆序这个栈，

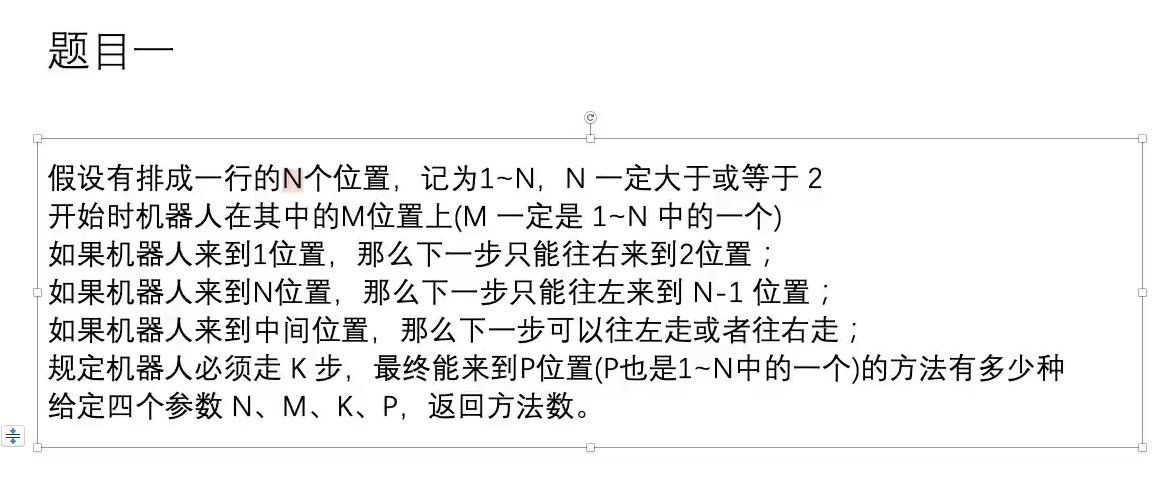
不能申请额外的数据结构，

只能使用递归函数。 如何实现?

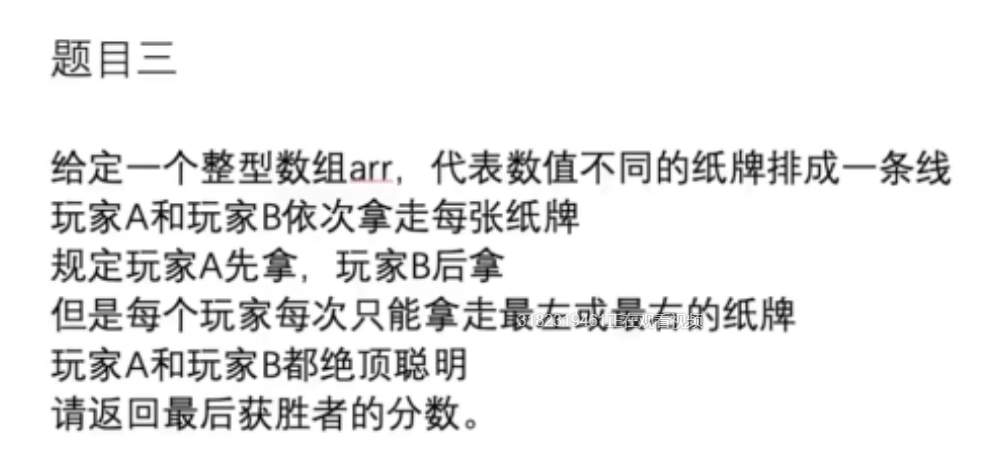
## 第18节 暴力递归到动态规划（一）

10:00

### √√机器人21:04



### √√纸牌1:28:25



## 第19节暴力递归到动态规划（二）

### 背包07:27

### √√1A 46:10

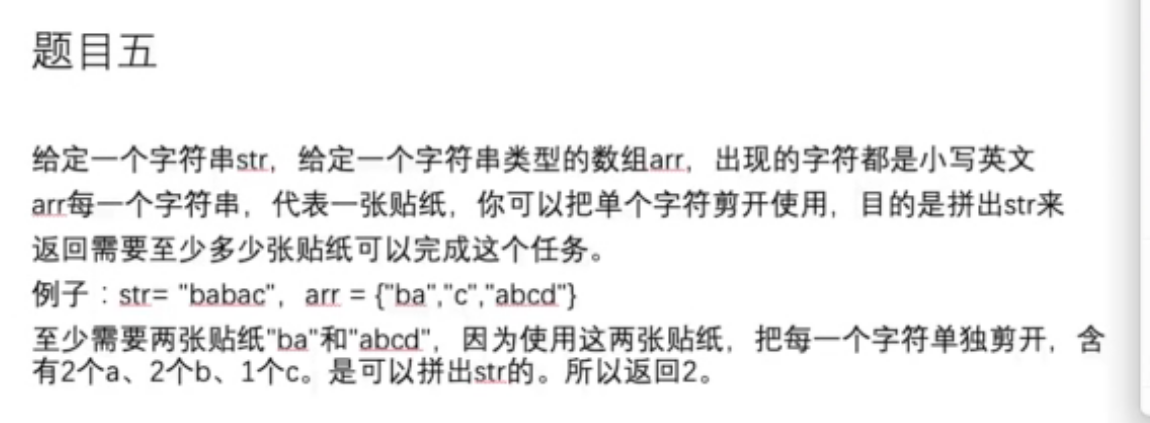
规定1和A对应、2和B对应、3和C对应...

那么一个数字字符串比如"111”就可以转化为:

"AAA"、"KA"和"AK"

给定一个只有数字字符组成的字符串str，返回有多少种转化结果

### √√贴纸1:10:05



### √√最长公共子序列1:55:16

## √第20节 暴力递归到动态规划（三）

### √√最长回文子序列12:00

1.原序列和反序列求公共子序列

2.四种可能性

动态规划

力扣516

### √√象棋问题1:01:34

1.递归

2.动态规划

### 咖啡机1:33:00

## √第21节 暴力递归到动态规划（四）

### √√最小路径和05:53

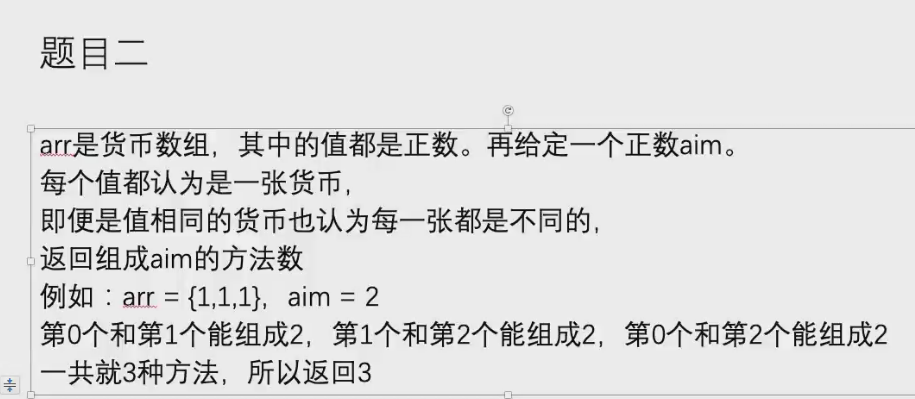
给定一个矩阵 m，从左上角开始每次只能向右或者向下走，

最后到达右下角的位置，路径上所有的数字累加起来就是路径和

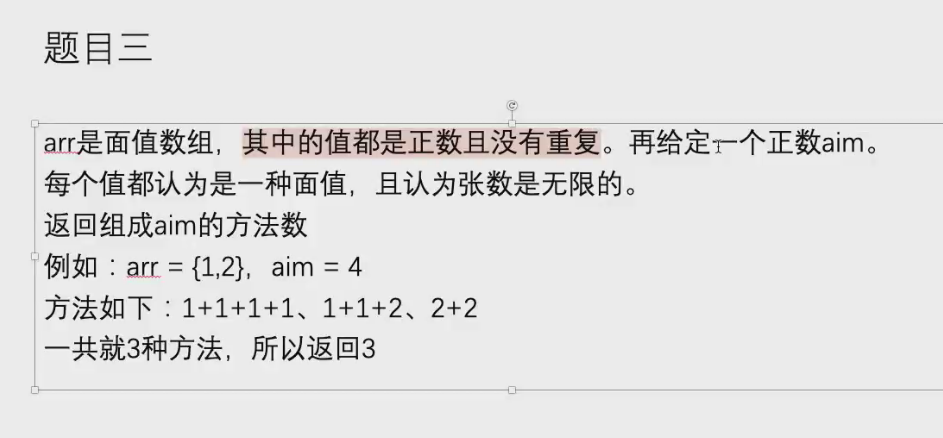
返回所有的路径中最小的路径和

数组压缩技巧

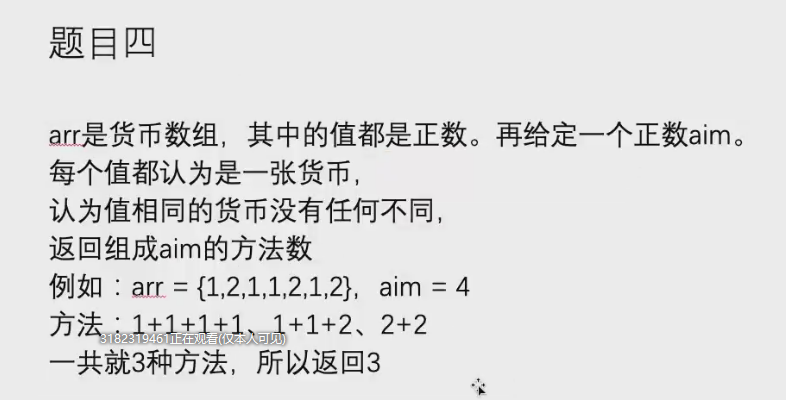
### 题目二 货币数组45:07



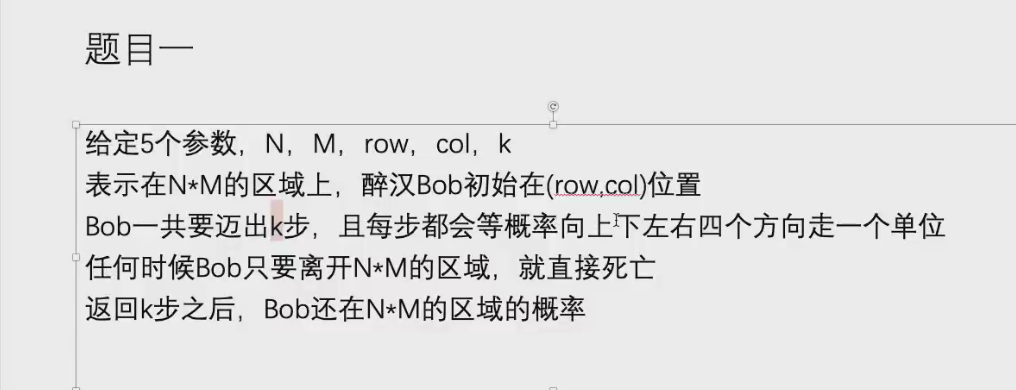
### 题目三 面值数组59:27



### 题目四 货币数组1:27:03

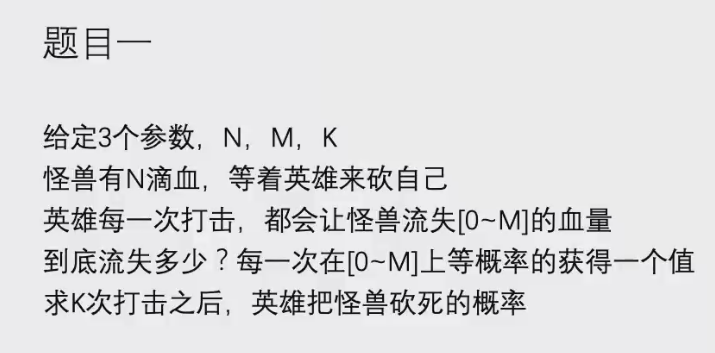


### 醉汉1:56:50

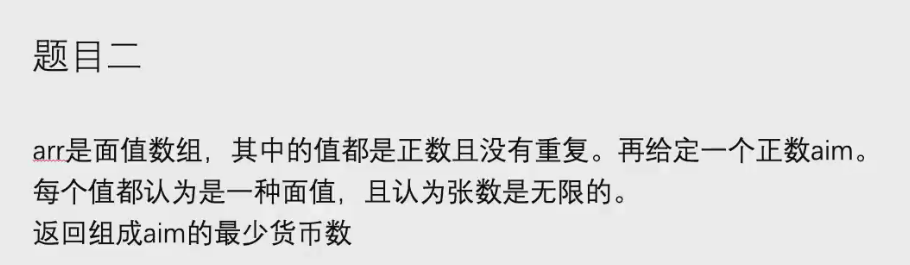


## √第22节 暴力递归到动态规划（五）

### 题目一 砍怪兽04:44



### √√不重复面值数组，无限张。 1:17:59



### √√一个数字裂开，5=2+3，升序02:03:22

给定一个正数n，求n的裂开方法数，

规定：后面的数不能比前面的数小

比如4的裂开方法有：

1+1+1+1、1+1+2、1+3、2+2、4

5种，所以返回5

有近似公式

## √第23节 暴力递归到动态规划（六）

### √题目一08：00

给定一个正数数组arr，

请把arr中所有的数分成两个集合，尽量让两个集合的累加和接近

返回：

最接近的情况下，较小集合的累加和

### √√题目二51:09

给定一个正数数组arr，请把arr中所有的数分成两个集合

如果arr长度为偶数，两个集合包含数的个数要一样多

如果arr长度为奇数，两个集合包含数的个数必须只差一个

请尽量让两个集合的累加和接近

返回：

最接近的情况下，较小集合的累加和

### √√n皇后1:52:44

N皇后问题是指在N\*N的棋盘上要摆N个皇后，

要求任何两个皇后不同行、不同列， 也不在同一条斜线上

给定一个整数n，返回n皇后的摆法有多少种。  
n=1，返回1

n=2或3，2皇后和3皇后问题无论怎么摆都不行，返回0

n=8，返回92

## √第24节 窗口内最大值或最小值的更新结构

### 滑动窗口07:30

### 双端队列15:33

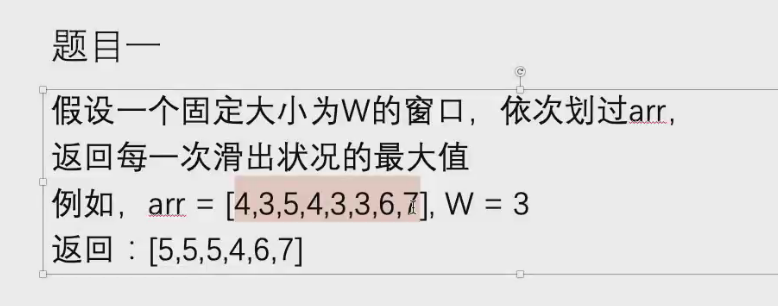
### √√题目一34:17

假设一个固定大小为W的窗口，依次划过arr，

返回每一次滑出状况的最大值

例如，arr = [4,3,5,4,3,3,6,7], W = 3

返回：[5,5,5,4,6,7]



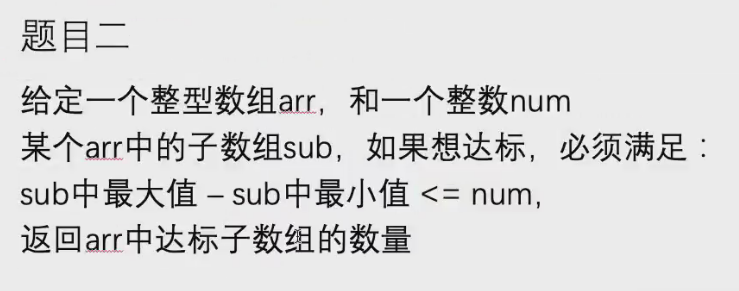
### √√题目二55:21

给定一个整型数组arr，和一个整数num

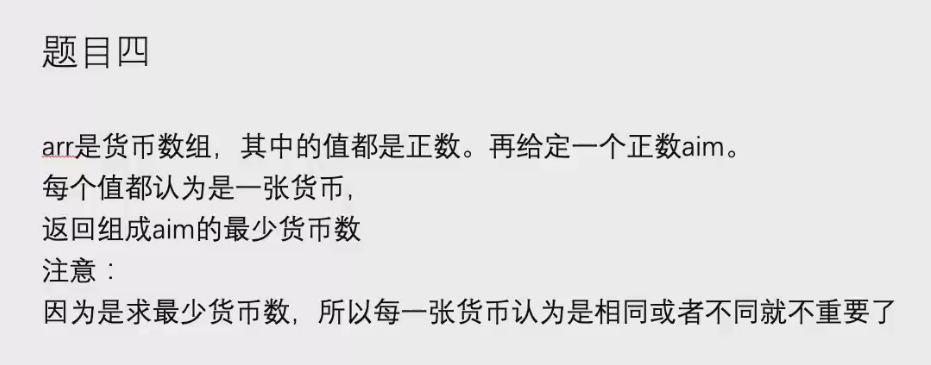
某个arr中的子数组sub，如果想达标，必须满足：

sub中最大值 – sub中最小值 <= num，

返回arr中达标子数组的数量



### 题目四2:01:41



### 134. 加油站，未讲

## 第25节 单调栈结构

### √单调栈5:00

一种特别设计的栈结构，为了解决如下的问题：

给定一个可能含有重复值的数组arr，i位置的数一定存在如下两个信息

1）arr[i]的左侧离i最近并且小于(或者大于)arr[i]的数在哪？

2）arr[i]的右侧离i最近并且小于(或者大于)arr[i]的数在哪？

如果想得到arr中所有位置的两个信息，怎么能让得到信息的过程尽量快。

那么到底怎么设计呢？

### √题目二50:17

给定一个只包含正数的数组arr，arr中任何一个子数组sub，

一定都可以算出(sub累加和 )\* (sub中的最小值)是什么，

那么所有子数组中，这个值最大是多少？

### √√直方图1:27:18

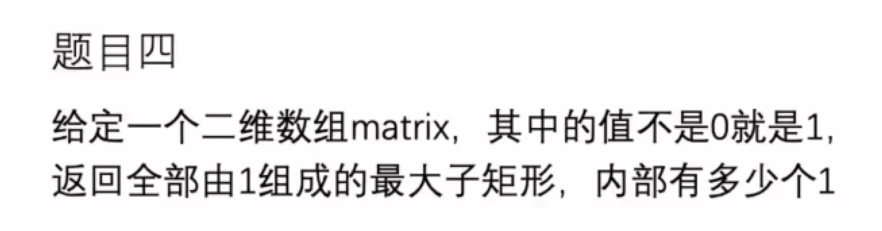
给定一个非负数组arr，代表直方图

返回直方图的最大长方形面积

### √√题目四1:34:47

给定一个二维数组matrix，其中的值不是0就是1，

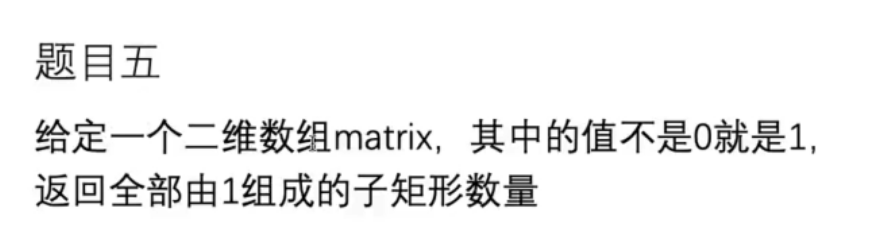
返回全部由1组成的最大子矩形，内部有多少个1



### √√题目五1:49:39未懂leetcode1504

给定一个二维数组matrix，其中的值不是0就是1，

返回全部由1组成的子矩形数量



## √第26节由斐波那契数列讲述矩阵快速幂技巧

### 题目一（单调栈续）6:14

给定一个数组arr，

返回所有子数组最小值的累加和

### 类似斐波那契数列的递归优化45:01

### f(n)+f(n-i)推广1:23:00

### √√题目四1:29:43

第一年农场有1只成熟的母牛A，往后的每年：

1）每一只成熟的母牛都会生一只母牛

2）每一只新出生的母牛都在出生的第三年成熟

3）每一只母牛永远不会死

返回N年后牛的数量

### √√题目五1:48:44

给定一个数N，想象只由0和1两种字符，组成的所有长度为N的字符串

如果某个字符串,任何0字符的左边都有1紧挨着,认为这个字符串达标

返回有多少达标的字符串

### 一个区域，只有两排1:59:55

## √第27节 bfprt算法、蓄水池算法

### √√KMP算法核心5:00

1）如何理解next数组

2）如何利用next数组加速匹配过程，优化时的两个实质！（私货解释）

## √第28节 Manacher算法

### √√Manacher算法01:22

假设字符串str长度为N，想返回最长回文子串的长度

时间复杂度O(N)

### √√添多少个字符，变成回文1:38:00

实际上是包含最后一个字符，最长回文子串1:38:44

### √√互为旋转子串1:51:14

第一个字符串相加，然后根据KMP看看第二个字符串是否是第一个字符串的子串

### √√两颗树，第二颗树是否是第一颗树的子树1:59:33

序列化，然后KMP

## 第29节

### √在无序数组中求第K小的数04:00

1）改写快排的方法

2）bfprt算法

### √给定一个无序数组arr中，给定一个正数k，返回top k个最大的数 1:09:51

不同时间复杂度三个方法：

1）O(N\*logN)

2）O(N + K\*logN)

3）O(n + k \* logk)

### √√蓄水池算法1:21:19

解决的问题：

假设有一个源源吐出不同球的机器，

只有装下10个球的袋子，每一个吐出的球，要么放入袋子，要么永远扔掉

如何做到机器吐出每一个球之后，所有吐出的球都等概率被放进袋子里

### uuid 1:58:38

## 第30节 Morris遍历

### 莫里斯遍历08:54

### √√给定一棵二叉树的头节点head，求以head为头的树中，最小深度是多少？ 1:23:00

## 第31节 线段树

### 线段树05:00

### 俄罗斯方块1:43:14

想象一下标准的俄罗斯方块游戏，X轴是积木最终下落到底的轴线

下面是这个游戏的简化版：

1）只会下落正方形积木

2）[a,b] -> 代表一个边长为b的正方形积木，积木左边缘沿着X = a这条线从上方掉落

3）认为整个X轴都可能接住积木，也就是说简化版游戏是没有整体的左右边界的

4）没有整体的左右边界，所以简化版游戏不会消除积木，因为不会有哪一层被填满。

给定一个N\*2的二维数组matrix，可以代表N个积木依次掉落，

返回每一次掉落之后的最大高度

## 第32节 IndexTree、AC自动机

### IndexTree 12:00

<https://shimo.im/docs/68cd6h3TwygPwx8W/read>

<https://shimo.im/docs/QxX9xy3VtTGPWpht/read> 1:17：02

### IndexTree2D 1:18:28

### AC1 1:33:00

### AC2 2:15:04

## 第33节与哈希函数有关的结构

### 哈希函数05:55

### 布隆过滤器42:44

### 一致性哈希1:20:40

## 第34节资源限制类题目的解题套路

### 资源限制技巧汇总01:49

1）布隆过滤器用于集合的建立与查询，并可以节省大量空间（已讲）

2）一致性哈希解决数据服务器的负载管理问题（已讲）

3）利用并查集结构做岛问题的并行计算（已讲）

4）哈希函数可以把数据按照种类均匀分流

5）位图解决某一范围上数字的出现情况，并可以节省大量空间

6）利用分段统计思想、并进一步节省大量空间

7）利用堆、外排序来做多个处理单元的结果合并

### 题目一14:23

32位无符号整数的范围是0~4,294,967,295，

现在有一个正好包含40亿个无符号整数的文件，

可以使用最多1GB的内存，怎么找到出现次数最多的数？

哈希均分成400份

### 题目二01:00:15

32位无符号整数的范围是0~4,294,967,295，

现在有一个正好包含40亿个无符号整数的文件，

所以在整个范围中必然存在没出现过的数。

可以使用最多1GB的内存，怎么找到所有未出现过的数？用位图

【进阶】

内存限制为 10MB，但是只用找到一个没出现过的数即可

### 题目三1:29:40

有一个包含100亿个URL的大文件，假设每个URL占用64B，

请找出其中所有重复的URL

【补充】

某搜索公司一天的用户搜索词汇是海量的(百亿数据量)，

请设计一种求出每天热门Top100词汇的可行办法

布隆过滤器

哈希取模

### 题目四1:31:27

32位无符号整数的范围是0~4294967295，

现在有40亿个无符号整数，

可以使用最多1GB的内存，

找出所有出现了两次的数。

### 题目五1:35:41

32位无符号整数的范围是0~4294967295，现在有40亿个无符号整数

可以使用最多10MB的内存，怎么找到这40亿个整数的中位数？

### 题目六1:45:56

32位无符号整数的范围是0~4294967295，

有一个10G大小的文件，每一行都装着这种类型的数字，

整个文件是无序的，给你5G的内存空间，

请你输出一个10G大小的文件，就是原文件所有数字排序的结果

## 第35节有序表（上）

### 搜索树11:00

### 平衡搜索树25:23

添加

删除

左无右无，直接删

左有右无，左直接挂载

左无右有，右直接挂载

左有右有，右树最左子节点，替换删除位置。左树最右子节点，替换删除位置。

LL：右旋

LR ：左子节点左旋，然后对 头节点 右旋。孙子上升到头节点了。

RL

RR：左旋

## 第36节有序表（中）

### sb树10:48

### 跳表1:09:49

## 第37节有序表（下）

### 题目一10:30

给定一个数组arr，和两个整数a和b（a<=b）

求arr中有多少个子数组，累加和在[a,b]这个范围上

返回达标的子数组数量

### 题目二1:06:24

有一个滑动窗口（讲过的）：

1）L是滑动窗口最左位置、R是滑动窗口最右位置，一开始LR都在数组左侧

2）任何一步都可能R往右动，表示某个数进了窗口

3）任何一步都可能L往右动，表示某个数出了窗口

想知道每一个窗口状态的中位数

### 题目三 有序表设计1:30:00

设计一个结构包含如下两个方法：

void add(int index, int num)：把num加入到index位置

int get(int index) ：取出index位置的值

void remove(int index) ：把index位置上的值删除

要求三个方法时间复杂度O(logN)

### 红黑树02:05:29

## 第38节 根据对数器找规律、根据数据量猜解法

### √√苹果16:30

### √√牛羊吃草31:53

### √√题目三54:00

定义一种数：可以表示成若干（数量>1）连续正数和的数

比如:

5 = 2+3，5就是这样的数

12 = 3+4+5，12就是这样的数

1不是这样的数，因为要求数量大于1个、连续正数和

2 = 1 + 1，2也不是，因为等号右边不是连续正数

给定一个参数N，返回是不是可以表示成若干连续正数和的数

### 不是题目1：01:20

### 题目四1:24:13

int[] d，d[i]：i号怪兽的能力

int[] p，p[i]：i号怪兽要求的钱

开始时你的能力是0，你的目标是从0号怪兽开始，通过所有的怪兽。

如果你当前的能力，小于i号怪兽的能力，你必须付出p[i]的钱，贿赂这个怪兽，然后怪兽就会加入你，他的能力直接累加到你的能力上；如果你当前的能力，大于等于i号怪兽的能力，你可以选择直接通过，你的能力并不会下降，你也可以选择贿赂这个怪兽，然后怪兽就会加入你，他的能力直接累加到你的能力上。

返回通过所有的怪兽，需要花的最小钱数。

## 第39节 根据数据量猜解法（续）、卡特兰数

### √√题目一12:17

给定一个非负数组arr，和一个正数m。

返回arr的所有子序列中累加和%m之后的最大值。

### 题目二46:00

### 卡特兰数1:25:46

### 括号1：38：00以前 1:43:35

### 家族1:40:00大概

### n个数1:59:00

### 股票上升45度2:02:19

### n个节点，多少种组成二叉树2:02:50 2:15:00

### 整数和偶数一样多2:09:04

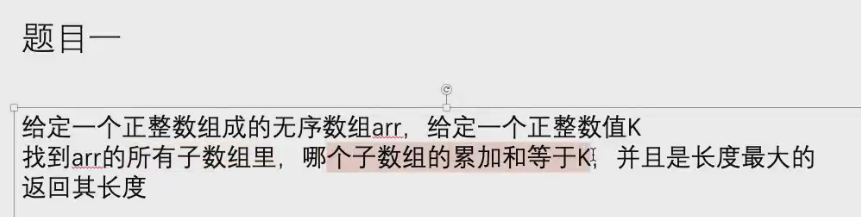
### 点一样多2:10:57

### 有限长度，无限个点2:13:20

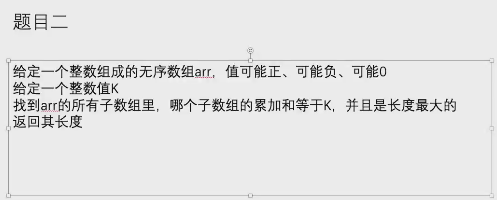
## 第40节 子数组达到规定累加和的最大长度系列问题

### 括号

### √√题目一17:00



### √√题目二 35：00



### √√子数组-1和1的数量54:37

无序数组arr，子数组-1和1的数量一样多，请问最长子数组的长度是多少？

### √√题目三56:32

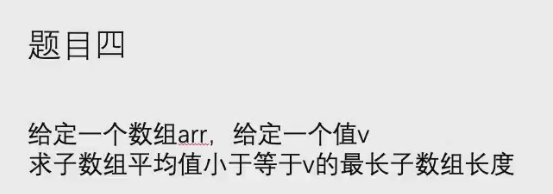
给定一个整数组成的无序数组arr，值可能正、可能负、可能0

给定一个整数值K

找到arr的所有子数组里，哪个子数组的累加和<=K，并且是长度最大的

返回其长度

### √√题目四1:48:14



### √√二维数组02:00:00

给定一个正方形矩阵matrix，原地调整成顺时针90度转动的样子

a b c g d a

d e f h e b

g h i i f c

<https://github.com/algorithmzuo/algorithmbasic2020/blob/master/src/class40/Code06_RotateMatrix.java>

### √√数组顺时针打印02:16:00

已经做过

### 打印星02:19:00

### √√反对角线打印02:24:00

## 第41节 四边形不等式技巧（上）

### √√题目一09:32

给定一个非负数组arr，长度为N，

那么有N-1种方案可以把arr切成左右两部分

每一种方案都有，min{左部分累加和，右部分累加和}

求这么多方案中，min{左部分累加和，右部分累加和}的最大值是多少？

整个过程要求时间复杂度O(N)

### 题目二16:55

把题目一中提到的，

min{左部分累加和，右部分累加和}，定义为S(N-1)，也就是说：

S(N-1)：在arr[0…N-1]范围上，做最优划分所得到的min{左部分累加和，右部分累加和}的最大值

现在要求返回一个长度为N的s数组，

s[i] =在arr[0…i]范围上，做最优划分所得到的min{左部分累加和，右部分累加和}的最大值

得到整个s数组的过程，做到时间复杂度O(N)

### √√题目三01:03:20

摆放着n堆石子。现要将石子有次序地合并成一堆

规定每次只能选相邻的2堆石子合并成新的一堆，

并将新的一堆石子数记为该次合并的得分

求出将n堆石子合并成一堆的最小得分（或最大得分）合并方案

### √√题目四01:44:44

给定一个整型数组 arr，数组中的每个值都为正数，表示完成一幅画作需要的时间，再 给定 一个整数 num，表示画匠的数量，每个画匠只能画连在一起的画作。所有的画家 并行工作，请 返回完成所有的画作需要的最少时间。

【举例】

arr=[3,1,4]，num=2。

最好的分配方式为第一个画匠画 3 和 1，所需时间为 4。第二个画匠画 4，所需时间 为 4。 因为并行工作，所以最少时间为 4。如果分配方式为第一个画匠画 3，所需时 间为 3。第二个画 匠画 1 和 4，所需的时间为 5。那么最少时间为 5，显然没有第一 种分配方式好。所以返回 4。

arr=[1,1,1,4,3]，num=3。

最好的分配方式为第一个画匠画前三个 1，所需时间为 3。第二个画匠画 4，所需时间 为 4。 第三个画匠画 3，所需时间为 3。返回 4。

## 第42节 四边形不等式技巧（下）

### 题目一21:28

一条直线上有居民点，邮局只能建在居民点上。给定一个有序正数数组arr，每个值表示 居民点的一维坐标，再给定一个正数 num，表示邮局数量。选择num个居民点建立num个 邮局，使所有的居民点到最近邮局的总距离最短，返回最短的总距离

【举例】

arr=[1,2,3,4,5,1000]，num=2。

第一个邮局建立在 3 位置，第二个邮局建立在 1000 位置。那么 1 位置到邮局的距离 为 2， 2 位置到邮局距离为 1，3 位置到邮局的距离为 0，4 位置到邮局的距离为 1， 5 位置到邮局的距 离为 2，1000 位置到邮局的距离为 0。这种方案下的总距离为 6， 其他任何方案的总距离都不会 比该方案的总距离更短，所以返回6

### 题目二1:04:08

一座大楼有 0~N 层，地面算作第 0 层，最高的一层为第 N 层。已知棋子从第 0 层掉落肯定 不会摔碎，从第 i 层掉落可能会摔碎，也可能不会摔碎(1≤i≤N)。给定整数 N 作为楼层数， 再给定整数 K 作为棋子数，返 回如果想找到棋子不会摔碎的最高层数，即使在最差的情况下扔 的最少次数。一次只能扔一个棋子。

【举例】

N=10，K=1。

返回 10。因为只有 1 棵棋子，所以不得不从第 1 层开始一直试到第 10 层，在最差的情况 下，即第 10 层 是不会摔坏的最高层，最少也要扔 10 次。

N=3，K=2。

返回 2。先在 2 层扔 1 棵棋子，如果碎了，试第 1 层，如果没碎，试第 3 层。 N=105，K=2 返回 14。

第一个棋子先在 14 层扔，碎了则用仅存的一个棋子试 1~13。 若没碎，第一个棋子继续在 27 层扔，碎了则 用仅存的一个棋子试 15~26。 若没碎，第一个棋子继续在 39 层扔，碎了则用仅存的一个棋子试 28~38。 若 没碎，第一个棋子继续在 50 层扔，碎了则用仅存的一个棋子试 40~49。 若没碎，第一个棋子继续在 60 层扔， 碎了则用仅存的一个棋子试 51~59。 若没碎，第一个棋子继续在 69 层扔，碎了则用仅存的一个棋子试 61~68。 若没碎，第一个棋子继续在 77 层扔，碎了则用仅存的一个棋子试 70~76。 若没碎，第一个棋子继续在 84 层 扔，碎了则用仅存的一个棋子试 78~83。 若没碎，第一个棋子继续在 90 层扔，碎了则用仅存的一个棋子试 85~89。 若没碎，第一个棋子继续在 95 层扔，碎了则用仅存的一个棋子试 91~94。 若没碎，第一个棋子继续 在 99 层扔，碎了则用仅存的一个棋子试 96~98。 若没碎，第一个棋子继续在 102 层扔，碎了则用仅存的一 个棋子试 100、101。 若没碎，第一个棋子继续在 104 层扔，碎了则用仅存的一个棋子试 103。 若没碎，第 一个棋子继续在 105 层扔，若到这一步还没碎，那么 105 便是结果。

## 第43节 状态压缩的动态规划

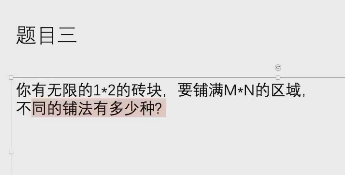


<https://shimo.im/docs/68cd6h3TwygPwx8W/read>

### 先后手10:00

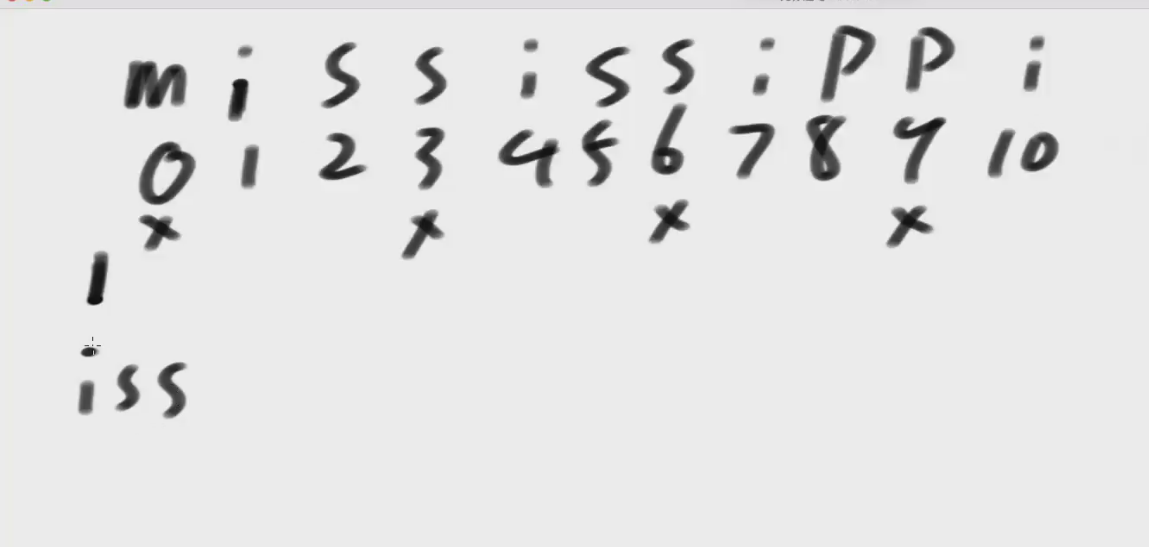
### √√tsp1:12:25

### 题目三1:52:35



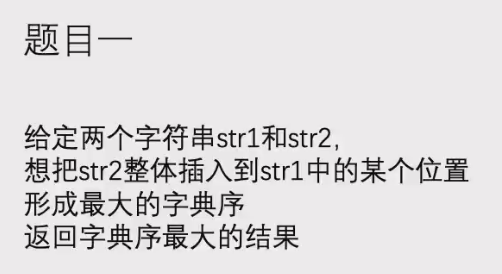
## √第44节 DC3生成后缀数组详解

### dc3

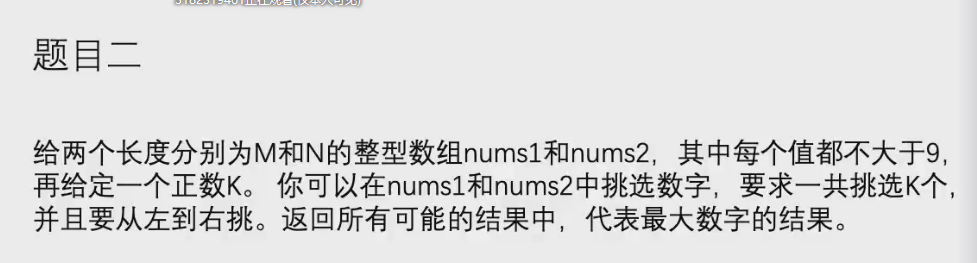


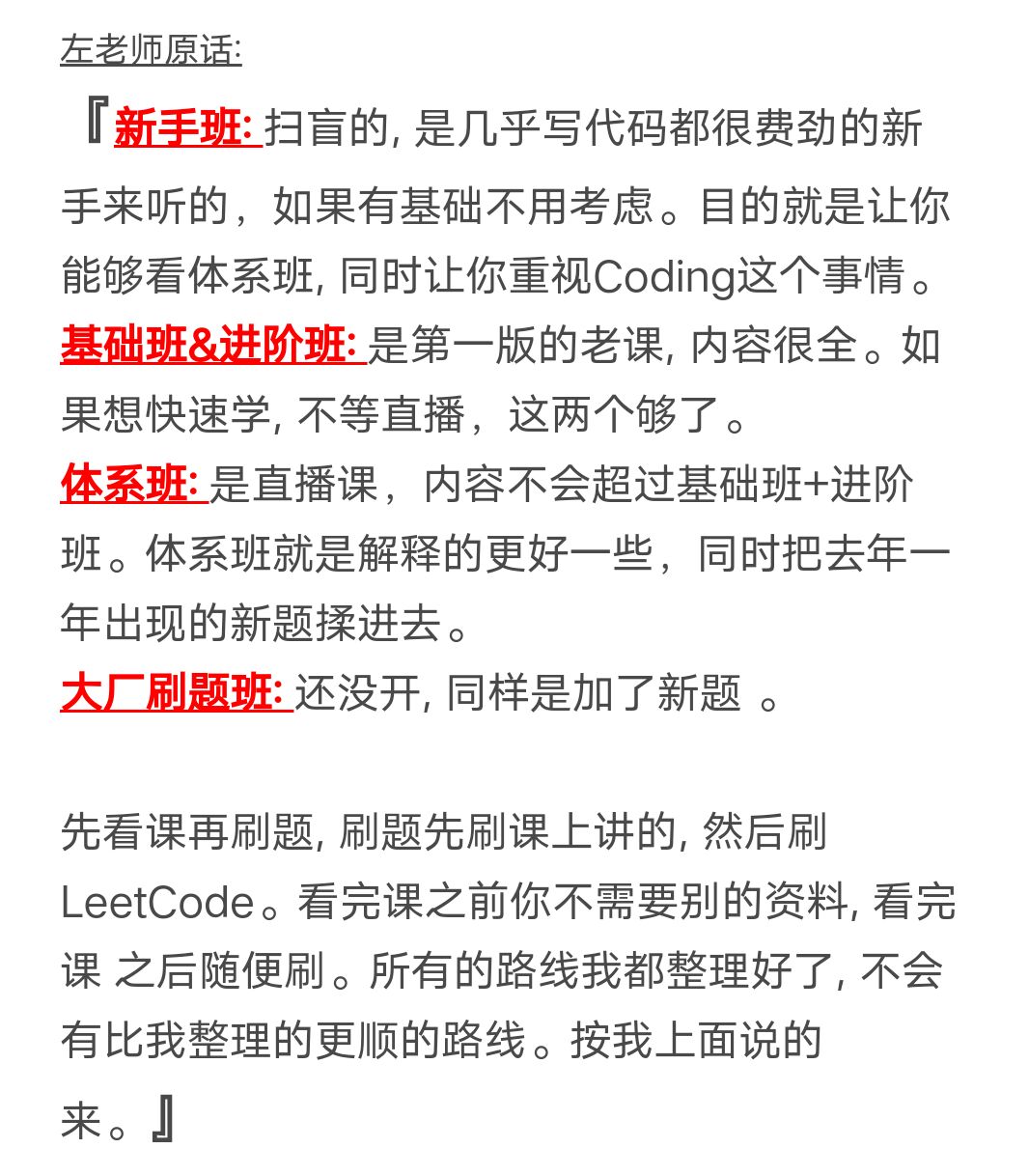
## 第45节 后缀数组解决的面试题

### √√题目一



### √√题目二41:49





## 第46节 动态规划猜法中和外部信息简化的相关问题（上）

### 题目一09:53

给定一个数组 arr，代表一排有分数的气球。每打爆一个气球都能获得分数，假设打爆气 球 的分数为 X，获得分数的规则如下: 1)如果被打爆气球的左边有没被打爆的气球，找到离被打爆气球最近的气球，假设分数为 L;如果被打爆气球的右边有没被打爆的气球，找到离被打爆气球最近的气球，假设分数为 R。 获得分数为 L\*X\*R。 2)如果被打爆气球的左边有没被打爆的气球，找到离被打爆气球最近的气球，假设分数为 L;如果被打爆气球的右边所有气球都已经被打爆。获得分数为 L\*X。 3)如果被打爆气球的左边所有的气球都已经被打爆;如果被打爆气球的右边有没被打爆的 气球，找到离被打爆气球最近的气球，假设分数为 R;如果被打爆气球的右边所有气球都 已经 被打爆。获得分数为 X\*R。 4)如果被打爆气球的左边和右边所有的气球都已经被打爆。获得分数为 X。

目标是打爆所有气球，获得每次打爆的分数。通过选择打爆气球的顺序，可以得到不同的总分，请返回能获得的最大分数。

【举例】

arr = {3,2,5} 如果先打爆3，获得3\*2;再打爆2，获得2\*5;最后打爆5，获得5;最后总分21 如果先打爆3，获得3\*2;再打爆5，获得2\*5;最后打爆2，获得2;最后总分18 如果先打爆2，获得3\*2\*5;再打爆3，获得3\*5;最后打爆5，获得5;最后总分50 如果先打爆2，获得3\*2\*5;再打爆5，获得3\*5;最后打爆3，获得3;最后总分48 如果先打爆5，获得2\*5;再打爆3，获得3\*2;最后打爆2，获得2;最后总分18 如果先打爆5，获得2\*5;再打爆2，获得3\*2;最后打爆3，获得3;最后总分19 返回能获得的最大分数为50

### 题目二01:01:38

<https://leetcode.com/problems/remove-boxes/>

### 题目三01:24:59

如果一个字符相邻的位置没有相同字符，那么这个位置的字符出现不能被消掉。比如:"ab"，其中a和b都不能被消掉

如果一个字符相邻的位置有相同字符，就可以一起消掉。比如:“abbbc”，中间一串的b是可以被消掉的，

消除之后剩下“ac”。某些字符如果消掉了，剩下的字符认为重新靠在一起

给定一个字符串，你可以决定每一步消除的顺序，目标是请尽可能多的消掉字符，返回最少的剩余字符数量

比如："aacca", 如果先消掉最左侧的"aa"，那么将剩下"cca"，然后把"cc"消掉，剩下的"a"将无法再消除，返回1

但是如果先消掉中间的"cc"，那么将剩下"aaa"，最后都消掉就一个字符也不剩了，返回0，这才是最优解。

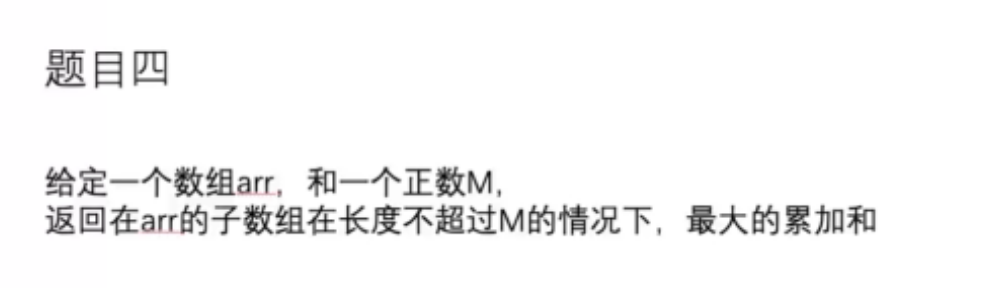
再比如："baaccabb"，如果先消除最左侧的两个a，剩下"bccabb"，如果再消除最左侧的两个c，剩下"babb"，

最后消除最右侧的两个b，剩下"ba"无法再消除，返回2

而最优策略是：先消除中间的两个c，剩下"baaabb"，再消除中间的三个a，剩下"bbb"，最后消除三个b，

不留下任何字符，返回0，这才是最优解

### 题目四01:47:45



### 哈夫曼树1:57:05

## 第47节 动态规划猜法中和外部信息简化的相关（下）

### 力扣664， 00:41

### 题目二54:44

整型数组arr长度为n(3 <= n <= 10^4)，最初每个数字是<=200的正数且满足如下条件：

1. arr[0] <= arr[1]

2.arr[n-1] <= arr[n-2]

3. arr[i] <= max(arr[i-1], arr[i+1])

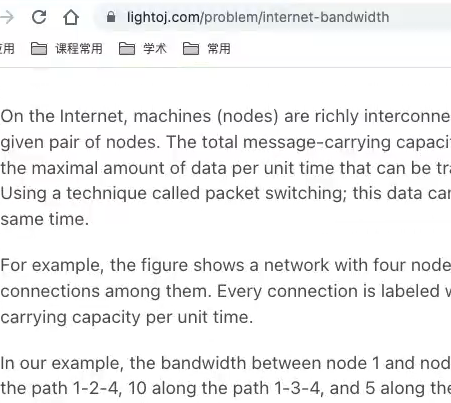
但是在arr有些数字丢失了，比如k位置的数字之前是正数，

丢失之后k位置的数字为0。

请你根据上述条件， 计算可能有多少种不同的arr可以满足以上条件。

比如 [6,0,9] 只有还原成 [6,9,9]满足全部三个条件，所以返回1种。

### 迪尼克算法，听声音是这个名字1:36:27 02:05:37代码



02:05:37