算法和数据结构体系学习班

00 算法和数据结构学前必看

内容：

算法和数据结构课程体系介绍

算法和数据结构学习路线

算法和数据结构学习方式推荐

算法和数据结构面试软技巧

同学们的各种答疑问题

01 时间复杂度、空间复杂度、对数器和二分法

内容：

评估算法优劣的核心指标

时间复杂度、空间复杂度、估算方式、意义

常数时间的操作

选择排序、冒泡排序、插入排序

最优解

对数器

二分法

题目：

选择排序及其对数器验证

冒泡排序及其对数器验证

插入排序及其对数器验证

有序数组中找到num

有序数组中找到>=num最左的位置

有序数组中找到<=num最右的位置

局部最小值问题

定义何为局部最小值：

arr[0] < arr[1]，0位置是局部最小；

arr[N-1] < arr[N-2]，N-1位置是局部最小；

arr[i-1] > arr[i] < arr[i+1]，i位置是局部最小；

给定一个数组arr，已知任何两个相邻的数都不相等，找到随便一个局部最小位置返回

02 异或运算、进一步认识对数器的重要性

内容：

异或运算的性质

异或运算的题目

题目：

不用额外变量交换两个数的值

不用额外变量交换数组中两个数的值

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

怎么把一个int类型的数，提取出二进制中最右侧的1来

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

一个数组中有一种数出现K次，其他数都出现了M次，

已知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)

03 单双链表、栈和队列、递归和Master公式、哈希表和有序表的使用和性能

内容：

单链表、双链表

栈、队列

递归的物理实质

评估递归复杂度的Master公式

哈希表的使用和性能

有序表的使用和性能

题目：

反转单链表、反转双链表

在链表中删除指定值的所有节点

用双链表实现栈和队列

用环形数组实现栈和队列

实现有getMin功能的栈

两个栈实现队列

两个队列实现栈

用递归行为得到数组中的最大值，并用master公式来估计时间复杂度

哈希表和有序表使用的code展示

04 归并排序及其常见面试题

内容：

归并排序

题目：

归并排序的递归和非递归实现

在一个数组中，一个数左边比它小的数的总和，叫该数的小和

所有数的小和累加起来，叫数组小和

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

给定一个数组arr，求数组小和

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，就称为降序对

给定一个数组arr，求数组的降序对总数量

在一个数组中，对于任何一个数num，求有多少个(后面的数\*2)依然<num，返回总个数

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

05 归并排序面试题(续)、快速排序

内容：

再来一个归并排序面试题

荷兰国旗问题

快速排序1.0

快速排序2.0

快速排序3.0

题目：

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

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

荷兰国旗问题的实现

快速排序从1.0到3.0的实现

快速排序的递归实现和非递归实现

code附加，双向链表进行快速排序的code实现

06 比较器、堆结构、堆排序

内容：

比较器

堆结构

堆排序

建立大根堆的两种方式，从上到下、从下到上，及其复杂度分析

题目：

比较器使用的code展示

堆结构的实现

堆排序的实现

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

k相对于数组长度来说是比较小的。请选择一个合适的排序策略，对这个数组进行排序。

07 和堆有关的面试题、加强堆结构

内容：

线段最大重合问题

加强堆的实现

题目：

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

加强堆的实现、注意点解析

做一个加强堆的题目，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…

一对arr[i]和op[i]就代表一个事件：

用户号为arr[i]，op[i] == T就代表这个用户购买了一件商品

op[i] == F就代表这个用户退货了一件商品

现在你作为电商平台负责人，你想在每一个事件到来的时候，

都给购买次数最多的前K名用户颁奖。

所以每个事件发生后，你都需要一个得奖名单（得奖区）。

得奖系统的规则：

1，如果某个用户购买商品数为0，但是又发生了退货事件，

则认为该事件无效，得奖名单和上一个事件发生后一致，例子中的5用户

2，某用户发生购买商品事件，购买商品数+1，发生退货事件，购买商品数-1

3，每次都是最多K个用户得奖，K也为传入的参数

如果根据全部规则，得奖人数确实不够K个，那就以不够的情况输出结果

4，得奖系统分为得奖区和候选区，任何用户只要购买数>0，

一定在这两个区域中的一个

5，购买数最大的前K名用户进入得奖区，

在最初时如果得奖区没有到达K个用户，那么新来的用户直接进入得奖区

6，如果购买数不足以进入得奖区的用户，进入候选区

7，如果候选区购买数最多的用户，已经足以进入得奖区，

该用户就会替换得奖区中购买数最少的用户（大于才能替换），

如果得奖区中购买数最少的用户有多个，就替换最早进入得奖区的用户

如果候选区中购买数最多的用户有多个，机会会给最早进入候选区的用户

8，候选区和得奖区是两套时间，

因用户只会在其中一个区域，所以只会有一个区域的时间，另一个没有

从得奖区出来进入候选区的用户，得奖区时间删除，

进入候选区的时间就是当前事件的时间（可以理解为arr[i]和op[i]中的i）

从候选区出来进入得奖区的用户，候选区时间删除，

进入得奖区的时间就是当前事件的时间（可以理解为arr[i]和op[i]中的i）

9，如果某用户购买数==0，不管在哪个区域都离开，区域时间删除，

离开是指彻底离开，哪个区域也不会找到该用户

如果下次该用户又发生购买行为，产生>0的购买数，

会再次根据之前规则回到某个区域中，进入区域的时间重记

请遍历arr数组和op数组，遍历每一步输出一个得奖名单

public List<List<Integer>> topK (int[] arr, boolean[] op, int k)

08 前缀树、不基于比较的排序（计数排序、基数排序）、排序算法的稳定性

内容：

前缀树

计数排序

基数排序

排序算法的稳定性

题目：

前缀树实现

计数排序

基数排序

09 排序算法大总结、链表及其相关面试题

内容：

排序算法总结

排序算法常见的坑

工程上对排序的常见改进

链表面试题的常见技巧

题目：

输入链表头节点，奇数长度返回中点，偶数长度返回上中点

输入链表头节点，奇数长度返回中点，偶数长度返回下中点

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

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

给定一个单链表的头节点head，给定一个整数n，将链表按n划分成左边<n、中间==n、右边>n

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

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)

10 链表相关面试题（续）、二叉树的常见遍历

内容：

单链表的相交节点系列问题

一种看似高效其实搞笑的节点删除方式

二叉树的中序、先序、后序遍历

题目：

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)

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？

二叉树先序、中序、后序的递归遍历和递归序

二叉树先序、中序、后序的非递归遍历

11 二叉树常见面试题和二叉树的递归套路（上）

内容：

通过题目来熟悉二叉树的解题技巧

题目：

二叉树的按层遍历

二叉树的序列化和反序列化

N叉树如何通过二叉树来序列化、并完成反序列化

Leetcode题目：https://leetcode.com/problems/encode-n-ary-tree-to-binary-tree/

打印二叉树的函数设计

求二叉树的最大宽度

求二叉树某个节点的后继节点

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

折纸问题

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。

N=1时，打印: down

N=2时，打印: down down up

12 二叉树常见面试题和二叉树的递归套路（中）

内容：

通过题目来熟悉二叉树的解题技巧

介绍二叉树的递归套路

1）假设以X节点为头，假设可以向X左树和X右树要任何信息

2）在上一步的假设下，讨论以X为头节点的树，得到答案的可能性（最重要）

3）列出所有可能性后，确定到底需要向左树和右树要什么样的信息

4）把左树信息和右树信息求全集，就是任何一棵子树都需要返回的信息S

5）递归函数都返回S，每一棵子树都这么要求

6）写代码，在代码中考虑如何把左树的信息和右树信息整合出整棵树的信息

题目：

判断二叉树是不是搜索二叉树

判断二叉树是不是平衡二叉树

判断二叉树是不是满二叉树

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

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

13 二叉树常见面试题和二叉树的递归套路（下）、贪心算法

内容：

二叉树递归套路继续实践

一道贪心算法从头到尾的完整做法

解决贪心题目的重要技巧，即对数器来验证脑洞

再次强调对数器的重要性

题目：

判断二叉树是不是完全二叉树（一般方法解决、递归套路解决）

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头节点

给定一棵二叉树的头节点head，和另外两个节点a和b，返回a和b的最低公共祖先

派对的最大快乐值

员工信息的定义如下:

class Employee {

public int happy; // 这名员工可以带来的快乐值

List<Employee> subordinates; // 这名员工有哪些直接下级

}

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

这个公司现在要办party，你可以决定哪些员工来，哪些员工不来，规则：

1.如果某个员工来了，那么这个员工的所有直接下级都不能来

2.派对的整体快乐值是所有到场员工快乐值的累加

3.你的目标是让派对的整体快乐值尽量大

给定一棵多叉树的头节点boss，请返回派对的最大快乐值。

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

14 贪心算法（续）、并查集

内容：

贪心算法继续实战

并查集详解

题目：

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

一块金条切成两半，是需要花费和长度数值一样的铜板

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

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

输入正数数组costs、正数数组profits、正数K和正数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：最后获得的最大钱数

并查集的实现

15 并查集相关的常见面试题

内容：

通过解答实际出现的面试题来体会并查集的优势、熟悉并查集的使用

题目：

一群朋友中，有几个不相交的朋友圈

Leetcode题目：https://leetcode.com/problems/friend-circles/

岛问题（递归解法 + 并查集解法 + 并行解法）

给定一个二维数组matrix，里面的值不是1就是0，上、下、左、右相邻的1认为是一片岛，返回matrix中岛的数量

16 图及其与图相关的算法

内容：

图的表达方式

图的常见描述

图的宽度优先遍历

图的深度优先遍历

图的拓扑排序

最小生成树算法Kruskal

最小生成树算法Prim

单元最短路径算法Dijkstra

题目：

图的数据结构抽象

实现图的宽度优先遍历

实现图的深度优先遍历

三种方式实现图的拓扑排序

用并查集实现Kruskal算法

用堆实现Prim算法

实现Dijkstra算法，用加强堆做更好的实现（16节+17节一开始）

17 用加强堆更好的实现Dijkstra算法、常见的递归

内容：

加强堆实现Dijkstra算法

递归的设计

常见的递归

题目：

打印n层汉诺塔从最左边移动到最右边的全部过程（递归+非递归实现）

打印一个字符串的全部子序列

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

打印一个字符串的全部排列

打印一个字符串的全部排列，要求不要出现重复的排列

给定一个栈，请逆序这个栈，不能申请额外的数据结构，只能使用递归函数

18 暴力递归到动态规划（一）

内容：

讲述暴力递归和动态规划的关系

记忆化搜索

动态规划都可以由暴力递归改进过来，解决动态规划的套路

常见的尝试模型

设计尝试过程的原则

本节是暴力递归到动态规划的总纲（很重要）

后续的课都是在讲述这一系列的套路

题目：

假设有排成一行的N个位置记为1~N，N一定大于或等于2

开始时机器人在其中的M位置上(M一定是1~N中的一个)

如果机器人来到1位置，那么下一步只能往右来到2位置；

如果机器人来到N位置，那么下一步只能往左来到N-1位置；

如果机器人来到中间位置，那么下一步可以往左走或者往右走；

规定机器人必须走K步，最终能来到P位置(P也是1~N中的一个)的方法有多少种

给定四个参数 N、M、K、P，返回方法数

给定一个整型数组arr，代表数值不同的纸牌排成一条线

玩家A和玩家B依次拿走每张纸牌

规定玩家A先拿，玩家B后拿

但是每个玩家每次只能拿走最左或最右的纸牌

玩家A和玩家B都绝顶聪明

请返回最后获胜者的分数

19 暴力递归到动态规划（二）

内容：

以18节为总纲

背包问题

记忆化搜索的一个很重要的注意点

通过面试题进一步强化动态规划的解题套路

题目：

背包问题

给定两个长度都为N的数组weights和values，weights[i]和values[i]分别代表 i号物品的重量和价值

给定一个正数bag，表示一个载重bag的袋子，装的物品不能超过这个重量

返回能装下的最大价值

规定1和A对应、2和B对应、3和C对应...26和Z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

给定一个字符串str，给定一个字符串类型的数组arr，出现的字符都是小写英文

arr每一个字符串，代表一张贴纸，你可以把单个字符剪开使用，目的是拼出str来

返回需要至少多少张贴纸可以完成这个任务

例子：str= "babac"，arr = {"ba","c","abcd"}

ba + ba + c 3 abcd + abcd 2 abcd+ba 2

所以返回2

给定两个字符串str1和str2，

返回这两个字符串的最长公共子序列长度

比如 ： str1 = “a12b3c456d”,str2 = “1ef23ghi4j56k”

最长公共子序列是“123456”，所以返回长度6

20 暴力递归到动态规划（三）

内容：

以18节为总纲

通过面试题进一步强化动态规划的解题套路

题目：

给定一个字符串str，返回这个字符串的最长回文子序列长度

比如 ： str = “a12b3c43def2ghi1kpm”

最长回文子序列是“1234321”或者“123c321”，返回长度7

请同学们自行搜索或者想象一个象棋的棋盘，

然后把整个棋盘放入第一象限，棋盘的最左下角是(0,0)位置

那么整个棋盘就是横坐标上9条线、纵坐标上10条线的区域

给你三个 参数 x，y，k

返回“马”从(0,0)位置出发，必须走k步

最后落在(x,y)上的方法数有多少种?

给定一个数组arr，arr[i]代表第i号咖啡机泡一杯咖啡的时间

给定一个正数N，表示N个人等着咖啡机泡咖啡，每台咖啡机只能轮流泡咖啡

只有一台咖啡机，一次只能洗一个杯子，时间耗费a，洗完才能洗下一杯

每个咖啡杯也可以自己挥发干净，时间耗费b，咖啡杯可以并行挥发

假设所有人拿到咖啡之后立刻喝干净，

返回从开始等到所有咖啡机变干净的最短时间

三个参数：int[] arr、int N，int a、int b

21 暴力递归到动态规划（四）

内容：

以18节为总纲

通过面试题进一步强化动态规划的解题套路

题目：

给定一个二维数组matrix，一个人必须从左上角出发，最后到达右下角

沿途只可以向下或者向右走，沿途的数字都累加就是距离累加和

返回最小距离累加和

arr是货币数组，其中的值都是正数。再给定一个正数aim。

每个值都认为是一张货币，

即便是值相同的货币也认为每一张都是不同的，

返回组成aim的方法数

例如：arr = {1,1,1}，aim = 2

第0个和第1个能组成2，第1个和第2个能组成2，第0个和第2个能组成2

一共就3种方法，所以返回3

arr是面值数组，其中的值都是正数且没有重复。再给定一个正数aim。

每个值都认为是一种面值，且认为张数是无限的。

返回组成aim的方法数

例如：arr = {1,2}，aim = 4

方法如下：1+1+1+1、1+1+2、2+2

一共就3种方法，所以返回3

arr是货币数组，其中的值都是正数。再给定一个正数aim。

每个值都认为是一张货币，

认为值相同的货币没有任何不同，

返回组成aim的方法数

例如：arr = {1,2,1,1,2,1,2}，aim = 4

方法：1+1+1+1、1+1+2、2+2

一共就3种方法，所以返回3

给定5个参数，N，M，row，col，k

表示在N\*M的区域上，醉汉Bob初始在(row,col)位置

Bob一共要迈出k步，且每步都会等概率向上下左右四个方向走一个单位

任何时候Bob只要离开N\*M的区域，就直接死亡

返回k步之后，Bob还在N\*M的区域的概率

22 暴力递归到动态规划（五）

内容：

以18节为总纲

通过面试题进一步强化动态规划的解题套路

斜率优化技巧

题目：

给定3个参数，N，M，K

怪兽有N滴血，等着英雄来砍自己

英雄每一次打击，都会让怪兽流失[0~M]的血量

到底流失多少？每一次在[0~M]上等概率的获得一个值

求K次打击之后，英雄把怪兽砍死的概率

arr是面值数组，其中的值都是正数且没有重复。再给定一个正数aim。

每个值都认为是一种面值，且认为张数是无限的。

返回组成aim的最少货币数

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

23 暴力递归到动态规划（六）

内容：

以18节为总纲

通过面试题进一步强化动态规划的解题套路

位信息技巧

题目：

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

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

24 窗口内最大值或最小值的更新结构

内容：

滑动窗口

窗口内最大值或最小值的更新结构

用题目来学习窗口内最大值或最小值的更新结构提供的便利性

题目：

窗口内最大值或最小值更新结构的实现

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]

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

加油站的良好出发点问题

动态规划中利用窗口内最大值或最小值更新结构做优化（难）

arr是货币数组，其中的值都是正数。再给定一个正数aim。

每个值都认为是一张货币，

返回组成aim的最少货币数

注意：因为是求最少货币数，所以每一张货币认为是相同或者不同就不重要了

25 单调栈

内容：

单调栈的原理（无重复数+有重复数）

用题目来学习单调栈提供的便利性

题目：

单调栈实现（无重复数+有重复数）

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？

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

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（面积）

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

26 单调栈相关的题目（续）、斐波那契数列的矩阵快速幂模型

内容：

再讲一个单调栈相关的面试题

斐波那契数列的矩阵快速幂模型详解

题目：

给定一个数组arr，返回所有子数组最小值的累加和

斐波那契数列矩阵乘法方式的实现

台阶方法数问题

一个人可以一次往上迈1个台阶，也可以迈2个台阶，返回迈上N级台阶的方法数

奶牛生小牛问题

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

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

用1\*2的瓷砖，把N\*2的区域填满，返回铺瓷砖的方法数

27 KMP算法

内容：

KMP算法

和KMP算法相关的面试题

题目：

KMP算法实现

给定两棵二叉树的头节点head1和head2，返回head1中是否有某个子树的结构和head2完全一样

判断str1和str2是否互为旋转字符串

28 Manacher算法

内容：

Manacher算法

和Manacher算法相关的面试题

题目：

Manacher算法实现

给定一个字符串str，只能在str的后面添加字符，想让str整体变成回文串，返回至少要添加几个字符

29 在无序数组中找到第K小的数、蓄水池算法

内容：

时间复杂度O(N)可以解决在无序数组中找到第K小的数，这个经典的面试题

改写快排的partition方法

bfprt算法

蓄水池算法

题目：

在无序数组中找到第K小的数（改写快排+bfprt）

设计在无序数组中收集最大的前K个数字的算法（根据不同的三个时间复杂度，设计三个算法）

给定一个无序数组arr中，长度为N，给定一个正数k，返回top k个最大的数

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)

蓄水池算法实现

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

30 二叉树的Morris遍历

内容：

二叉树之前的遍历方式有空间浪费的问题

Morris遍历时间复杂度O(N)，额外空间复杂度O(1)，通过利用原树中大量空闲指针的方式，达到节省空间的目的

假设来到当前节点cur，开始时cur来到头节点位置

1）如果cur没有左孩子，cur向右移动(cur = cur.right)

2）如果cur有左孩子，找到左子树上最右的节点mostRight：

a.如果mostRight的右指针指向空，让其指向cur，

然后cur向左移动(cur = cur.left)

b.如果mostRight的右指针指向cur，让其指向null，

然后cur向右移动(cur = cur.right)

3）cur为空时遍历停止

Morris遍历实现二叉树的先序、中序、后序遍历

题目：

Morris遍历的实现

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？

31 线段树

内容：

线段树是一种支持范围整体修改和范围整体查询的数据结构

线段树解决的问题范畴：大范围信息可以只由左、右两侧信息加工出，而不必遍历左右两个子范围的具体状况

题目：

给定一个数组arr，用户希望你实现如下三个方法

1）void add(int L, int R, int V) : 让数组arr[L…R]上每个数都加上V

2）void update(int L, int R, int V) : 让数组arr[L…R]上每个数都变成V

3）int sum(int L, int R) :让返回arr[L…R]这个范围整体的累加和

怎么让这三个方法，时间复杂度都是O(logN)

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

Leetcode题目：https://leetcode.com/problems/falling-squares/

32 IndexTree、AC自动机

内容：

IndexTree

1）支持区间查询

2）没有线段树那么强，但是非常容易改成一维、二维、三维的结构

3）只支持单点更新

AC自动机

解决在一个大字符串中，找到多个候选字符串的问题

1）把所有匹配串生成一棵前缀树

2）前缀树节点增加fail指针

3）fail指针的含义：如果必须以当前字符结尾，当前形成的路径是str，剩下哪一个字符串的前缀和str的后缀

拥有最大的匹配长度。fail指针就指向那个字符串的最后一个字符所对应的节点（迷不迷？听讲述！）

题目：

IndexTree在一维数组和二维数组上的实现

AC自动机的实现

33 与哈希函数有关的结构

内容：

哈希函数

哈希函数的应用

布隆过滤器

一致性哈希

题目：

原理讲述为主，面试只会聊设计，所以本节无题目

34 资源限制类题目的解题套路

内容：

布隆过滤器用于集合的建立与查询，并可以节省大量空间

一致性哈希解决数据服务器的负载管理问题

利用并查集结构做岛问题的并行计算

哈希函数可以把数据按照种类均匀分流

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

利用分段统计思想、并进一步节省大量空间

利用堆、外排序来做多个处理单元的结果合并

题目：

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？

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？

进阶：内存限制为 3KB，但是只用找到一个没出现过的数即可

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

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

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3K的内存，怎么找到这40亿个整数的中位数？

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

35 有序表（上）

内容：

平衡搜索二叉树

左旋

右旋

AVL树的节点违规4种类型（LL，LR，RL，RR）

题目：

AVL树的实现

36 有序表（中）

内容：

size-balanced-tree详解

skiplist详解

聊聊红黑树

题目：

size-balanced-tree实现

skiplist实现

37 有序表（下）

内容：

讲解有序表相关的面试题

讲解改写有序表的题目核心点

题目：

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

有一个滑动窗口：

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

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)

假设有打乱顺序的一群人站成一个队列，数组people表示队列中一些人的属性（不一定按顺序）

每个people[i]=[hi, ki]表示第i个人的身高为hi，前面正好有ki个身高大于或等于hi的人

请你重新构造并返回输入数组people所表示的队列，返回的队列应该格式化为数组queue

其中queue[j]=[hj, kj]是队列中第j个人的属性（queue[0] 是排在队列前面的人）。

Leetcode题目：https://leetcode.com/problems/queue-reconstruction-by-height/

38 根据对数器找规律、根据数据量猜解法

内容：

讲解对数器找规律的解题技巧

讲解根据数据量猜解法的技巧

1）C/C++，1秒处理的指令条数为10的8次方

2）Java等语言，1~4秒处理的指令条数为10的8次方

3）这里就有大量的分析提示了

题目：

小虎去买苹果，商店只提供两种类型的塑料袋，每种类型都有任意数量

1）能装下6个苹果的袋子

2）能装下8个苹果的袋子

小虎可以自由使用两种袋子来装苹果，但是小虎有强迫症，他要求自己使用的袋子数量必须最少，

且使用的每个袋子必须装满，给定一个正整数N，返回至少使用多少袋子。如果N无法让使用的每个袋子必须装满，返回-1

给定一个正整数N，表示有N份青草统一堆放在仓库里，有一只牛和一只羊，牛先吃，羊后吃，它俩轮流吃草

不管是牛还是羊，每一轮能吃的草量必须是：1，4，16，64…(4的某次方)

谁最先把草吃完，谁获胜，假设牛和羊都绝顶聪明，都想赢，都会做出理性的决定。根据唯一的参数N，返回谁会赢

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

2=1+1，2不是这样的数，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

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

（课上会给出不同的数据量描述）

39 根据数据量猜解法（续）、分治技巧、卡特兰数

内容：

继续熟悉根据数据量猜解法

讲解分治法

讲解卡特兰数（课上证明的时候有小错，在40节开始处修正了）

题目：

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

牛牛家里一共有n袋零食, 第i袋零食体积为v[i]，背包容量为w，牛牛想知道在总体积不超过背包容量的情况下,

一共有多少种零食放法，体积为0也算一种放法

1 <= n <= 30, 1 <= w <= 2 \* 10^9，v[I] (0 <= v[i] <= 10^9）

假设给你N个0，和N个1，你必须用全部数字拼序列，返回有多少个序列满足任何前缀串，1的数量都不少于0的数量

有N个二叉树节点，每个节点彼此之间无任何差别，返回由N个二叉树节点，组成的不同结构数量是多少？

题目补充： arr中的值可能为正，可能为负，可能为0，自由选择arr中的数字，能不能累加得到sum（多种做法）

40 子数组达到规定累加和的最大长度系列问题、矩阵处理技巧题

内容：

修正了39节卡特兰数讲解时的一个小错误

熟悉子数组达到规定累加和的三个模型（正、有正有负有0、累加和<=K）

矩阵处理技巧的宏观调度coding技巧

题目：

给定一个正整数组成的无序数组arr，给定一个正整数值K，找到arr的所有子数组里，哪个子数组的累加和等于K

并且是长度最大的，返回其长度

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等于K，并且是长度最大的，返回其长度

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

给定一个数组arr，给定一个值v，求子数组平均值小于等于v的最长子数组长度

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

给定一个正方形或者长方形矩阵matrix，实现转圈打印

给定一个正方形或者长方形矩阵matrix，实现zigzag打印

转圈打印星号\*问题

41 四边形不等式技巧（上）

内容：

区间划分问题中的划分点不回退现象

四边形不等式技巧特征

1，两个可变参数的区间划分问题

2，每个格子有枚举行为

3，当两个可变参数固定一个，另一个参数和答案之间存在单调性关系

4，而且两组单调关系是反向的：(升 升，降 降) (升 降，降 升)

5，能否获得指导枚举优化的位置对：上+右，或者，左+下

四边形不等式技巧注意点

1，不要证明！用对数器验证！

2，枚举的时候面对最优答案相等的时候怎么处理？用对数器都试试！

3，可以把时间复杂度降低一阶

O(N^3) -> O(N^2)

O(N^2 \* M) -> O(N \* M)

O(N \* M^2) -> O(N \* M)

4，四边形不等式有些时候是最优解，有些时候不是

不是的原因：尝试思路，在根儿上不够好

题目：

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)

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)

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

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，返回完成所有的画作需要的最少时间

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

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

42 四边形不等式技巧（下）

内容：

继续熟悉四边形不等式

展示好的尝试是最关键的

题目：

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

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

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

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

43 状态压缩的动态规划

内容：

动态规划的状态压缩技巧

题目：

在"100 game"这个游戏中，两名玩家轮流选择从1到10的任意整数，累计整数和

先使得累计整数和达到或超过100的玩家，即为胜者，如果我们将游戏规则改为 “玩家不能重复使用整数” 呢？

例如，两个玩家可以轮流从公共整数池中抽取从1到15的整数（不放回），直到累计整数和 >= 100

给定一个整数 maxChoosableInteger （整数池中可选择的最大数）和另一个整数 desiredTotal（累计和）

判断先出手的玩家是否能稳赢（假设两位玩家游戏时都表现最佳）

你可以假设 maxChoosableInteger 不会大于 20， desiredTotal 不会大于 300。

Leetcode题目：https://leetcode.com/problems/can-i-win/

TSP问题

有N个城市，任何两个城市之间的都有距离，任何一座城市到自己的距离都为0

所有点到点的距离都存在一个N\*N的二维数组matrix里，也就是整张图由邻接矩阵表示

现要求一旅行商从k城市出发必须经过每一个城市且只在一个城市逗留一次，最后回到出发的k城

参数给定一个matrix，给定k。返回总距离最短的路的距离

铺砖问题（最优解其实是轮廓线dp，但是这个解法对大厂刷题来说比较难，掌握课上的解法即可）

你有无限的1\*2的砖块，要铺满M\*N的区域，

不同的铺法有多少种?

44 DC3生成后缀数组详解

内容：

后缀数组

介绍用DC3算法生成后缀数组的流程

题目：

给你一个字符串 s，找出它的所有子串并按字典序排列，返回排在最后的那个子串

Leetcode题目：https://leetcode.com/problems/last-substring-in-lexicographical-order/

DC3算法的实现（完全根据论文描述）

45 后缀数组解决的面试题

内容：

通过题目进一步熟悉DC3算法

通过DC3算法得到height数组

题目：

给定两个字符串str1和str2，想把str2整体插入到str1中的某个位置，形成最大的字典序，返回字典序最大的结果

给两个长度分别为M和N的整型数组nums1和nums2，其中每个值都不大于9，再给定一个正数K。 你可以在nums1和nums2中挑选数字，要求一共挑选K个，并且要从左到右挑。返回所有可能的结果中，代表最大数字的结果

最长公共子串问题是面试常见题目之一，假设str1长度N，str2长度M

一般在面试场上回答出O(N\*M)的解法已经是比较优秀了

因为得到O(N\*M)的解法，就已经需要用到动态规划了

但其实这个问题的最优解是O(N+M)，需要用到后缀数组+height数组

课上将对本题解法代码进行详解

46 动态规划猜法中和外部信息简化的相关问题（上）、哈夫曼树

内容：

以18节做总纲

有些动态规划面试题，需要很好的设计参数，这种设计方式都有"外部信息简化"的特征

哈夫曼树

题目：

有n个气球，编号为0到n-1，每个气球上都标有一个数字，这些数字存在数组nums中

现在要求你戳破所有的气球。戳破第i个气球，你可以获得nums[i - 1] \* nums[i] \* nums[i + 1] 枚硬币

这里的i-1和i+1代表和i相邻的、没有被戳爆的！两个气球的序号

如果i-1或i+1超出了数组的边界，那么就当它是一个数字为1的气球

求所能获得硬币的最大数量

Leetcode题目：https://leetcode.com/problems/burst-balloons/

给出一些不同颜色的盒子，盒子的颜色由数字表示，即不同的数字表示不同的颜色，你将经过若干轮操作去去掉盒子

直到所有的盒子都去掉为止，每一轮你可以移除具有相同颜色的连续k个盒子（k >= 1）

这样一轮之后你将得到 k \* k 个积分，当你将所有盒子都去掉之后，求你能获得的最大积分和

Leetcode题目：https://leetcode.com/problems/remove-boxes/

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：

如果先消除中间的两个c，剩下"baaabb"，如果再消除中间的三个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

给定一个数组arr，和一个正数M，返回在arr的子数组在长度不超过M的情况下，最大的累加和

哈夫曼树的实现

47 动态规划猜法中和外部信息简化的相关问题（下）、最大网络流算法之Dinic算法

内容：

进一步解决带有"外部信息简化"特征的动态规划

Dinic算法

题目：

有台奇怪的打印机有以下两个特殊要求：

打印机每次只能打印由同一个字符组成的序列。

每次可以在任意起始和结束位置打印新字符，并且会覆盖掉原来已有的字符。

给你一个字符串s，你的任务是计算这个打印机打印它需要的最少打印次数。

Leetcode题目：https://leetcode.com/problems/strange-printer/

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. 0位置的要求：arr[0]<=arr[1]

2. n-1位置的要求：arr[n-1]<=arr[n-2]

3. 中间i位置的要求：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，即[6,9,9]达标

Dinic算法详解

测试链接：https://lightoj.com/problem/internet-bandwidth