251 Flatten 2D Vector

Review III

251-1 如果一层或多层为空怎么办。设计题一定要考虑打破原有的（层级）数据结构.

252-2 Sol 1 如果想保持list的结构，不要设置totallevel totalinlist 因为某一层可能为空。根据discussion的答案，只保留index list和index element即可。

252 Meeting Room

253 Meeting Room II  
Review III

253-1 sol 1画图考虑变化。每当一个meeting结束会有什么发生

253-2 sol 2 另外考虑，会议室用interval考虑，放在queue中

254 Factor Combinations

255 Verify Preorder Sequence In BST

Review III

255-1 Stack用于tree的遍历。当发现当前值大于前一个值的时候一定到了右孩子。

256 Paint House

Review III

256-1 DP based on a group of data on each level!!! Revisit!!!

257 BT Paths

258 AddDigits

Review II

258-1 如何把Integer变成String：

String curstr = curnum + ””;

String curstr = String.valueOf(curnum);

259 3Sum Smaller

Review III

259-1 双指针站肩可以解决大于小于等于的问题并且降低维度！！！

259-2 可以有负数的情况。所以不能限定最右或者最左的值

259-3 Binary search 和 two pointer解法往往同命相连，请看leetcode solution

260 Single Number III

ReviewIII

260-1 revisit single number II:

<http://www.cnblogs.com/higerzhang/p/4159330.html>

其中方法三实际上是对三十二位数同时进行运算

260-2 如何思考本题：要找到两个数，所以如果能分两组，每组分别做抑或肯定能的结果。所以找的一个set为1的位数对整个数组进行平分即可。

How to get the last set digit 1:

Num& ~(num-1)

261 Graph Valid Tree

261-1 满足两点即可：第一 没有环，第二所有点都要通过neighbor遍历到 没有遗漏

261-2 Union Find to resolve the edges problem。可以设置 parent[] 和rank[] 两个数组。其实有一个隐含的条件：如果本题成立，edge的length必然是n-1（每个edge union一次，最后所有点只有一个parent）

262 Trips and Users

263 Ugly Number

264 Ugly Number II

Review III

264-1 DP based on three pointers

265 Paint House II

Review III

265-1 DP the current value based on the pre-group of values. We can keep more than 2 values as the DP references in each level

266 267 Palindrome Permutation:

Review III

266-1 大写和小写字母属于不同情况，请分开考虑。

266-2 Set可以用java8提供的 for(char c: set) 遍历

268 Missing Number

268-1 Number continuous problem always think about “Mathematics solution”

268-2 数字和坐标一一对应关系，不要忘了可以使用抑或。

269 Alien Dictionary

Review III

Topological Sort!!!

269-1 有的元素可能出现1次或多次，有的元素可能根本就不会出现。所以在考虑dependency的数组的时候，0代表这个元素从来没有出现过。1为出现过（至少一次）。当有dependency的时候都会大于等于2。

269-2 把所有dependency为1的元素放入queue。每次出queue即计入结果。然后找邻居。当邻居的dependency降为1 的时候 入queue

269-3 topological sort成功的结果是所有元素最后都能入queue。一旦出现环，count!=result.length()

270 Closest BST

271 Encoding and Decoding

271-1 在输入每一段字符之前先输入其长度和#

271-2 另一种解法是不输入长度只输入#。如果字符串里面已经有#怎么办？double#。所以一见到double的#则一同处理为字符串内的内容。一见到单个的#（后面不再有#了）便会知道是不同的字符串。

272 Closest BST Value II

Review III

272-1 Closest有两个方向。所以如果从上到下DFS的左右都需要遍历（无法判断右孩子的极端左孩子或者左孩子的极端右孩子哪个更近）所以O(n)是少不了的了。

272-2 解法1: 既然O(n)少不了，则考虑inorder (preorder) traverse. 分隔开大于target的和小于target的用两个stack储存起来再双指针比较。

273 Integer to English Words

Review III

273-1 String[] array不能声明为final。因为String是object！！！

如何声明String数组String[] digit = {"","One","Two”,”Three”};

273-2 对于某些String需要重复引用多次的情况，用String[] 数组，注意角标和元素内容的一一对应。

274 275 H-Index

ReviewII

274-1 如何实现在数组中O(n)的解法对数组元素进行分类：另造一个数组，利用角标和元素(Integer值)的对应关系。

275-1 当题中有条件说，数组已经sorted好，需要找某个元素的的时候，一定注意可以考虑Binary Search从而优化O(n) 🡪 O(logn)

276 Find the Celebrity

276-1 My Solution: Greedy, dynamic decide the next input in recursion

277 Find the Celebrity

Review II

277-1 Think again and again on loop 1!!!

278 First Bad Version

Review III

278-1 My solution : start < end -1 cannot pass. Please figure out why! Thanks!

279 Perfect Squares

279-1 对角标进行操作的DP

280 Wiggle Sort

280-1 根据坐标奇偶特性决定是否要swap

281 Zigzag Iterator

281-1 hasNext() function should have considered all the possibilities.

282 Expression Add Operators

Review II

282-1 乘法需要keep一个pre才可以做。

282-2 注意用subString在String里面取数

282-3 数字越界问题，可以用Long.parseLong()

283 Move Zeroes

284 Peek Iterator

Review II

284-1 因为永远需要知道next的值。所以next的update只在next() 函数中进行。 hasNext()只负责查询next变量是否有值

285 Inorder Successor BST

Review II

285-1 从recursion本身的意思上去理解！

286 Walls and Gates

Review II

286-1 我的做法非常好。再看一遍！

287