

codeforces round 656(div3)

A Three Pairwise Maximums

题意：给定 x, y, z ，且 $x = \max(a, b), y = \max(a, c), z = \max(b, c)$ ，求符合条件的 a, b, c

题解：这题说实话一开始给我搞懵了，仔细想了想之后发现其实挺简单，首先可以发现，若 x, y, z 中至少有两个相同，而且相同的两个数都是 a, b, c 中的最大数，然后就十分简单了，取 a 为最大数， b 为另一个数， c 取1即可

B Restore the Permutation by Merger

题意：将两个排列合并在一起（可穿插），给合并后的序列，问原排列是什么样。

题解：是排列，则每个数至多出现1次，枚举每个数，这个数第二次出现时候直接输出即可。

C Make It Good

题意：要将一个序列通过移走一系列前缀变成一个 good string 一个 good string 定义为可行如下操作成为空串：每次从串头或者串尾拿走一个数放入一个新序列，最后这个新序列可以成为一个不下降序列。

题解：看了题目里的描述就会明白，目标串就是一个山坡状起伏的串，先上升再下降，从后往前找第二个 $a_{i+1} < a_i$ 的值遍可。

D a-Good String

题意：难以描述，上链接<https://codeforces.com/contest/1385/problem/D>

题解：其实是一个暴力dfs的问题，我走了一小时弯路，还是太菜了.....，正解就是枚举最后串的形态，暴力dfs比对两个字符串即可。

E Directing Edges

题意：给了一张图，其中有些边是已经确定方向的，有些边是没有确定方向的，现在要给那些没有确定方向的边确定一个放向，是其成为一个有向无环图。

题解：有向无环图，首先想到拓扑排序，这题偏向结论性，在拓扑排序后，如果有一条从 u 到 v 的边，则拓扑排序后 v 一定在 u 之后。所以这题我们先对所有点及有向边进行拓扑排序，存点的小顺序，然后按照顺序连边即可。

F Removing Leaves

题意：给定一颗树和一个数字 k ，每次要经行一个操作：去除通过一个父节点的 k 个叶节点。问最多能经行几次操作？

题解：其实是一个贪心的问题，记录每个存在子节点为叶节点的节点，按照子叶节点个数从大到小排序，每次取出第一个，后去除这些叶节点，再放入集合中，这需要开一个重定序的set来维护，然后用`vector<set<int>`来维护每个点的叶节点和与之相连的所有点即可，要注意这个过程中还要维护普通点在删边过程中会向叶节点转换。

G Columns Swaps

题解：给定两排数，每排都有 n 个数，每次操作可以交换一列的两个数，问能否存在一个最少的交换方案，使操作之后每一行都是 1 到 n 的一个排列。

题解：这题有点难想，大致是一个二分图的问题。首先遍历两行数，如果一个数的出现次数超过了2次，那么肯定不成立。如果所有数出现次数都是两次，那么一定有一种方案满足。现定四个数组 $r1[n], r2[n], b1[n], b2[n], b$ 数组用于存放列数， r 数组用于存放行数，如果对于一个数 i ， $b_{1i} = b_{2i}$ ，则不考虑这个点，因为肯定不会动这个点的。如果 $b_{1i} \neq b_{2i}$ ，考虑 r 数组，如果 $r_{1i} \neq r_{2i}$ ，则在

From:
<https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:manespace:codeforces_round_656_div3&rev=1595562178

Last update: 2020/07/24 11:42

