20240528题解

A

一个桶就搞定了。

B

直接枚举原串的前k位和后7-k位即可。

C

如果sum(A[i])<sum(B[i])直接输出无解。否则贪心。

先让数组A的元素报告它们与数组B对应的元素多/少多少。sum记录下来一共还剩下多少，同时把有贡献（指多的）的贡献值记录在一个vector【vector记作v】里面

定义一个计数器count，赋值为n，把所有的贡献值从小往大排序，然后遍历，如果v[i]<=sum就让sum-=v[i]且count--，退出循环时输出count即可。

D

计数题，比较考查思维。

发现如果从大到小填数，由于后面要填的数都比前面已经填好的数小，所以之后不管怎么填都不会影响每一行和每一列的最大值。

下面为方便描述，称一行或一列被覆盖，当这一行或列已经填了至少一个数。

设当前已经被覆盖的行数和列数分别为 nn 和 mm，记答案为 ans，对当前要填的每个数进行分类讨论：

有一行和一列以它为最大值，此时填的位置固定；

有一行或一列以它为最大值，此时它只能填在那一行或一列一个被覆盖的位置上，ans←ans×nn 或 ans←ans×mm；

没有行和列以它为最大值，此时它只能填在一个行列都被覆盖的位置上，即一个交点上，ans←ans×(nn×mm−al)，其中 al 表示当前已经填了多少数。

时间复杂度O(nm)，可以通过本题。

E

有n个点，第i个点的权值为ai。给定常数d，在点i,j之间连边的代价是∣i−j∣d+ai+aj​。求将它们连成一棵树的最小代价。

经典的完全图最小生成树，但是看数据范围，直接Prim和Kruskal都不行。

因此需要减少边数。

考虑将所有点对半折开，左半部分和右半部分的建边分治处理，只考虑横跨两部分的建边，

若左半部分的点为 j，右半部分的点为 i，即j<i，边权就可以转换成：ai+i\*d+aj-j\*d

因此我们只需要选出：

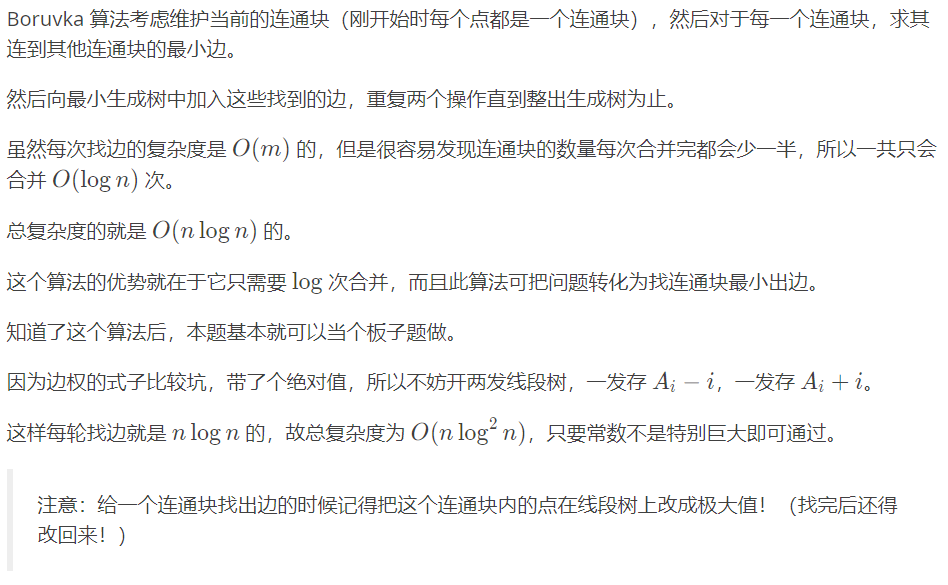
右半部分 min{Ai+i×d} 所对应的 i，

左半部分 min{Aj−j×d} 所对应的 j，

将j和右半边的所有点连边，i和左半边的所有点连边。

因此边数规模降为O(n logn)，然后Kruskal即可，时间复杂度O(n log2n)。

当然直接Boruvka也可以，这样的话复杂度是O(n logn)，更好一点。



F

