

2020/07/25-2020/07/31周报

团队训练

李元恺

题目

比赛

[M-SOLUTIONS Programming Contest 2020\(atcoder\)](#) pros:5/6/6 rk:249

姜维翰

比赛

[cf edu 92 码农专场.jpg](#)

袁熙

专题

没有专题

比赛

没有比赛

题目

补题（板子）牛客第五场B

[链接](#)

题意：给边带权的树，可以连边或删边，要求始终连通且形成的环异或和=0，求最后图的最小权值和
思路：对原图的树，可以把边权用所连点的权值异或和来表示，转化成异或最小生成树

类似的模板题：[链接](#)

本周推荐

李元恺

[Codeforces Round 659 1C](#)

标签：图论、状压dp

题意：两个字符串S和T，字符集大小为20，每次可以从S中选任意多个相同字符，将它们变成另一个字符，问最小多少次可以将S变成T。

思路：将每个字符作为一个点，如果存在字符 $s_i = a$ 且 $t_i = b$ 建立一条 $a \rightarrow b$ 的有向边，计这个图为 G_1 。

1、观察可以发现，如果一个弱联通图的最优解不是链状（若所有操作都满足上一次操作选择的边的终点等于下一次操作的起点，则是链状），则一定存在一个操作数相同的链状方案。

证明：显然每个点都至少要被一条边覆盖，如果不是链状，则一定有两条边指向相同点，此时可以把先操作的边的目标改为后操作的边的起点，重复这样调整，最终会得到一个操作数相同的链状解。

2、现在我们可以用一个序列 $\{a_n\}$ 来表示答案 $[a_1, a_2, \dots, a_n]$ 表示依次执行 $a_1 \rightarrow a_2, a_2 \rightarrow a_3, \dots, a_{n-1} \rightarrow a_n$

序列 a_1, a_2, \dots, a_n 合法 iff 任意 G_1 中的边 $u \rightarrow v$ 存在 $i < j$ 满足 $i < j$ 且 $a_i = u, a_j = v$

3、对于 G_1 的弱联通块 G ，计其无环导出子图中点数最多的为 S ，计 $|S|=k$ ， $|G|=n$ ，则答案大于等于 $2n-k-1$

证明：设有一个解的大小为 k ，考虑向一个空图中依次加入 k 条边。

1) 若某次操作的两个端点处于同一个连通块中，则对于任意无环导出子图 DAG ，其大小最多减1，连通块数不变。

2) 若不处于同一连通块中，则连通块数-1，所有DAG体积不变。

初始状态下有 n 个连通块，所以最多 $k-n+1$ 条边被用来减小DAG的体积，最开始我们有一个大小为 n 的DAG，所以最终最大的DAG体积至少是 $2*n - k - 1$

4、存在一个使答案等于 $2n-k-1$ 的方案，给出构造：设 $G(V, E)$ 最大DAG导出子图 $= S(V_1, E_1)$ ，令序列为 $V \setminus V_1, V_1, V \setminus V_1$

5、状压dp求出最大DAG导出子图

comment: 这周做的最有意思的题，推荐给大家

袁熙

[CF1372F 题目链接](#)

标签：构造 交互

题意：一个长为 n 不降的序列，由 k 个不同的数组成，每次可以询问 $[l, r]$ 中的众数（数量相同取更小值），要求在 $4k$ 次询问内确定原序列

思路：考虑如果已知某个数在某个位置 p 和这个数的总出现次数 f 那么可以用两次询问 $(p, p+f-1), (p-f+1, p)$ 得到这个数的结果。那么如果能使所有数都出现在查询结果中，就能知道整个序列。要想构造出上述的查询，我们可以从 $(1, n)$ 开始，用当前区间的最高频率 f 为间隔，每次确认 $f, 2f, \dots$ 这些点的数值，这些数值两两不等，且 $(1, n)$ 查询得到的众数一定在这些数值中。接下来递归地对两边的区间进行询问。这样对于每个数，找到它的一个出现位置+确定它的出现区间+递归时确定它作为众数的区间，共 $4k$ 次询问。

comment[]做法比较多的构造题

姜维翰

CF1292C

DP题

题意：给一棵 n 个点的树，要求给每条边选择一个 0 到 $n-2$ 的数，不可以重复。 $\text{mex}(u, v)$ 表示点 u 到点 v 经过的数的 mex 求最大化所有点对的 mex 和， $n=3000$

解法：可以看到，树上存在一条链，链上包含 $[0, l]$ 上的所有值，答案只和这个链的位置和链上数的摆法有关。我们还可以通过贪心，看出链应该是这个样子的： $10, 8, 6, 4, 2, 0, 1, 3, 5, 7, 9$ 于是链的摆法就确定了

那么我们就 $f[u][v]$ 来表示链为 u 到 v 时的答案

在算答案的时候要把 mex 求和转化成对 mex 大于某个值的点对的计数

这样就可以 n^2 的时间内完成了

comment[]很独特的一道dp题，答案计数方法的转换也很经典

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

Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:acm_life_from_zero:7.25-7.31&rev=1596181581

Last update: 2020/07/31 15:46

