

团队训练

比赛时间	比赛名称	当场过题数	至今过题数	总题数	排名
2020-08-01	2020牛客暑期多校第七场	4	6	10	66/1090
2020-08-03	2020牛客暑期多校第八场	4	6	11	32/685
2020-08-06	2020-2021 BUAA ICPC Team Supplementary Training 02	6	8	10	6/19

本周推荐

2sozx

Codeforces 804D Expected diameter of a tree

- 分类：树形DP，根号分治
- 题意：给定一片森林， q 此询问，每次给出两个点 u, v 。如果 u, v 在一棵树内输出 -1 ，否则在这两棵树任取一点临时建立一条边，求连边后的直径的期望。 $n, q \leq 10^5$
- 题解：首先我们可以预处理出每个点在哪棵树中，其次预处理出每个点 u 到这棵树叶子的最大值 $mx[u]$ 。这个可以用树形DP处理，将每棵树按照这个最大值进行排序，最后在处理出每棵树的直径长度 len 。询问的时候枚举点数少的树，在另一棵树中寻找另一个点。将两棵树连接 u, v 后的直径有两种情况： $mx[u] + mx[v] + 1$ 和 $\max(len[u], len[v])$ 。第二种情况是一个定值，因此对于每一个 v 我们可以二分出满足第一种情况的 u 的个数，剩余的即为第二种情况。最后答案要用 map 记录下来避免重复询问。复杂度是神奇的 $O(n\sqrt{n}\log n)$
- comment：根号分治太神奇了

Bazoka13

Codeforces 1083E The Fair Nut and Rectangles

- 分类：dp
- 题意：给定 n 个带有权值的第一象限的矩形，并且每个矩形有两条边与坐标轴重合，选择一个子集，使得子集内的矩形面积并减去权值和的差最大
- 题解：按照横坐标排序后，可以写出一个 dp 转移式： $dp_i = x_i * y_i - a_i + \max(-x_j * y_i + dp_j)$ 。而 \max 后面的式子可以通过凸包来维护，为了熟悉板子的使用换了插入直线和查询单点最大值的做法，需要注意由于可能存在负数，需要插入 $(0, 0)$
- comment：dp 的转移式的推导比较巧妙，之后就变成裸题了，（为什么会放在 1E 啊草）

JJLeo

2020牛客多校第七场I Valuable Forests

- 分类：prufer序列，组合计数。
- 题意：\$n\$个不同的点的生成森林中，每个点权值为该点的度数和平方，问所有生成森林的所有点的权值和是多少？\$T\$组数据\$(n, T \le 5000)\$
- 题解：每个点都是对称的，因此只需固定一个点最后乘以\$n\$即可。首先设\$h_i\$为\$i\$个不同的点的生成森林数量，利用prufer序列可以得到\$O(n^2)\$递推式
$$h_i = \sum_{j=0}^{i-1} \binom{i-1}{j} j^{i-j-1}$$
接下来设\$g_i\$为我们所考虑的点所在的树大小为\$i\$时所有情况该点贡献的权值和，考虑将该点固定为根，然后枚举该点的度数，利用prufer序列算出方案数，再乘以度数的平方和，得到\$O(n^2)\$递推式
$$g_i = \sum_{j=1}^{i-1} \binom{i-2}{j-1} (i-1)^{i-2-(j-1)}$$
最后设\$f_i\$为节点数为\$i\$时一个固定节点对答案的贡献，考虑枚举该点所在树的大小，则剩下的节点组成森林，可以得到\$O(n^2)\$递推式
$$f_i = \sum_{j=2}^i \binom{i-1}{j-1} g_{j-1} f_{i-j}$$
对于每个询问，我们只需\$O(1)\$输出\$nf_n\$即可。
- comment：对prufer序列的高级应用，十分巧妙。

2020牛客多校第八场H Hard String Problem

- 分类：字符串。
- 题意：给出\$n\$个字符串，将每个字符串重复循环无限次，问得到的\$n\$个新字符串有多少个本质不同的公共子串，若有无限个输出\$-1\$。\$(\sum |s_i| \le 3 \times 10^5)\$
- 题解：首先求出每个字符串的最小表示法（使用Lyndon分解，C++自带rotate()也可以很好实现循环移位），将每个字符串用其最短循环节表示（求出next数组，设\$x\$为最后一个位置的next数组值，字符串长度为\$len\$，若\$(len-x)|len\$则最小循环节长度为\$len-x\$，否则最小循环节长度为\$len\$）。如果出现\$n\$个相同的字符串，显然有无穷多个公共子串；否则由弱周期引理可以证明，对于两个字符串来说公共子串长度不超过长串的三倍，对于多个字符串来说，只需考虑最短串和其它串的公共子串的交即可。
- 具体来说，将除最短串外的串扩大到四倍，将最短串扩大到最长串的四倍，然后求这些字符串的公共子串数目即可。使用SAM来求过于繁琐还容易写锅，对于多串问题可以使用更为简便的广义SAM，只需将所有串插入广义SAM，然后对于每个串，将所有能到达的点标记，然后考虑所有标记数为\$n\$的节点即可。标记时只需按照字符串走一遍，每走到一个点不断跳link直到跳到的点已经被该点标记为止，这样每个字符串相当于标记了属于自己的SAM，而SAM节点数量是线性的，因此总复杂度为\$O(n)\$。
- comment：如果不看结论，可以说是结合了多种字符串算法，非常综合。

题目

- [每日亿题2020.8.2](#)
- [每日亿题2020.8.7](#)

个人训练

2sozx

比赛

- 摸了

题目

- 每日亿题

Bazoka13

比赛

意外情况，摸了

题目

JJLeo

比赛

题目

From:

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

Permanent link:

https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:farmer_john:week_14&rev=1596794973 

Last update: **2020/08/07 18:09**