

2020.08.08-2020.08.14 周报

团队训练

_wzx27

专题

[Matrix Exponentiation](#)

题目

牛客九

[A - Groundhog and 2-Power Representation](#) | [G - Groundhog Chasing Death](#)

牛客十

[D - Hearthstone Battleground](#)

比赛

Infinity37

专题

暂无。

题目

牛客九

[F - Groundhog Looking Dowdy](#) | [K - The Flee Plan of Groundhog](#)

牛客十

[E - Game](#) | [J - Identical Trees](#)

比赛

暂无。

Zars19

专题

无。

题目

比赛

- Codeforces Round #662 (Div. 2)
- Codeforces Round #663 (Div. 2)
- Codeforces Round #664 (Div. 1)
- AtCoder Grand Contest 047

本周推荐

Infinity37

来源 [luoguP3296](#)

tag:树hash [费用流转移dp](#)

概述 [□](#)

给定一颗树和两套01权值，现在可以花费1的代价修改某点的权值，问最小修改几次可以使得第一套权值和第二套权值的树同构。

答案 [□](#)

先找到重心，以重心为根对树进行hash [□](#)如果有两个重心那就增加一个重心连接两个重心再进行树hash [□](#)

我们设状态 $F_{i,j}$ 代表第一套权值的 i 子树与第二套权值的 j 子树同构的最小代价。具体转移要使用一个二分图完备匹配的费用流，对 i,j 这两棵树的所有子树 [□](#)hash值相同并且树高相同的连接一条边，我们假设这两个点是 u,v [□](#)这条边的流量为1，费用为 $F_{u,v}$ [□](#)然后依次转移即可。

comments:费用流转移dp的另一道题目，和第十场的题目主要区别在于无根树的处理，找到重心进行树hash

[_wzx27](#)

来源 [AGC047C](#)

tag [FFT](#) [原根](#)

概述

给出 n 个数，两两乘积模 P 的和。

答案

如果从类似生成函数的角度入手，把贡献累计在 x^i 的指数 i 上就可以用 FFT 求出贡献。但注意到只把指数作为模 P 的值会有问题，由于 $x^i \cdot x^j = x^{i+j}$ 但我们实际想要的是 $i*j$ 的答案。于是考虑原根的对数来表示就可以转化成乘法了。

先预处理出每个原根的幂次 $g^{i \% P}$ 的值和 $x = g^{i \% p}$ 的对应幂次 i ，然后做一次 FFT 即可，但注意要减去同一个数乘了自己的贡献。

comments：一个原根的经典用法，需要掌握。

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

Permanent link: https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:wangzai_milk:weekly15&rev=1597396064 

Last update: 2020/08/14 17:07