

图论 & 数据结构

袁浩天

IIIS, Tsinghua

2023.8.2

目录

① 专题题目选讲

100

$$n \leq 10^5, 1 \leq a_i \leq 10^5$$

对于 $d > \sqrt{n}$, 注意到这种情况下等差数列一定只在一个很小的范围内, 我们直接暴力对相邻元素建图找最长路即可。

Code

阅读 CF1654E.cpp

CF 1555F

你需要维护一个 n 个点的无向带权图，初始为空。称一个图是好的当且仅当它的所有简单环的边权异或和为 1。

支持 Q 次询问，每次给出一条新的边 (u, v, w) ，如果这条边加入之后图仍然是好的就加入，否则不加入。你需要快速回答是否加入。

$$n, Q \leq 10^5$$

Solution

通过观察我们能够发现一个性质：一个好的图中任意两个简单环的公共部分最多只有一个顶点。换句话说这张图是一个边仙人掌。

Solution

通过观察我们能够发现一个性质：一个好的图中任意两个简单环的公共部分最多只有一个顶点。换句话说这张图是一个边仙人掌。接着我们考虑怎么维护这个仙人掌

Solution

首先我们先求出所有生成森林里的边，这些边是不会受到环边的影响的。也就是说目前所有的树结构已经固定了。

考虑加入一条环边 (u, v) ，首先需要 check 树上路径异或和环边权值的异或是否为 1，这个只需要预处理到根的异或值即可。然后我们需要查询是否路径上的边是否有被使用过。

两种不同的做法：1. 直接 HLD 用线段树维护 2. 我们注意到路径也是可以减的所以直接维护到根的路径就行了，同时由于每条边最多被使用一次可以暴力一条一条修改。

Code

阅读 CF1555F.cpp

题目描述

$$n \leq 70000, m \leq 150000, w, h \leq n, t_i \leq 10000$$

KD-Tree

KD-Tree 是线段树在高维空间的一个扩展，能够在低于线性的复杂度内支持高维的区间修改/区间查询。

类似的, KD-Tree 也是一个二叉树结构, 以二维为例每个点维护一个矩形 $[lx_v, rx_v] \times [ly_v, ry_v]$, 一般来说这个矩形会对应一个点集, 所以一般会选择包含整个点集的最小矩形。

如果矩形内包含了不止一个点那么这个点的两个儿子会按照 x 轴或者 y 轴将空间平均划分成两个矩形，如此迭代。

区间修改和查询的方式与线段树相同，在复杂度分析上一般使用 $n^{\frac{k-1}{k}}$ 来估计。

Solution

首先直接暴力建图是不行的，那么不难想到直接使用 KDT 进行优化建图。KDT 帮助我们将一个任意的矩形拆成了 $O(\sqrt{n})$ 个节点。

如果直接建图空间复杂度是 $O(m\sqrt{n})$ ，无法接受。但是我们可以考虑不显式建图跑 dijkstra，具体流程如下：

- 设置两类节点，一类普通的点 $1 - n$ ，另一类表示一个矩形内的点。维护堆 Q 。
- 在 dijkstra 的过程中我们用 KDT 维护当前还没有被弹出的点的集合。
- 每次弹出点的时候，如果这个点是一个普通点那么我们直接根据它的转移更新若干矩形点并插入堆。
- 如果弹出的点是矩形点那我们就直接通过 KDT 遍历这个矩形里尚未被弹出的普通点进行普通点的弹出和更新。

CF 938G

给定无向连通带权图, 定义一条路径的长度为路径上边权的异或, 如果在一条路径上经过了多次那么在异或的时候也会被计算多次。

你需要支持一下三种操作:

- 加入一条连接 x 和 y 的长度为 d 的边, 保证加入这条边之前 x 和 y 之间没有边。
- 删除链接 x 和 y 的边, 保证删除之前这条边存在且删除边之后图仍然连通。
- 计算 x 到 y 的最短路径, 路径可以不是简单路径。

$n, m, q \leq 200000$

Solution

考虑对询问建一个线段树，每条边存在一个加入时间 l_i 和一个删除时间 r_i 。那么它会对所有 $[l_i, r_i]$ 区间上的边产生贡献，把这条边打在线段树的懒标记上。那么每个三号询问本质上对应了一个线段树的叶子，它对应图的边就是叶子到根的懒标记的并。

Solution

考虑对询问建一个线段树，每条边存在一个加入时间 l_i 和一个删除时间 r_i 。那么它会对所有 $[l_i, r_i]$ 区间上的边产生贡献，把这条边打在线段树的懒标记上。那么每个三号询问本质上对应了一个线段树的叶子，它对应图的边就是叶子到根的懒标记的并。考虑对线段树做一个 DFS，进入一个节点的时候加上这个点的贡献，走出一个节点的时候 rollback 贡献即可。

Solution

思考这些信息需要什么数据结构来维护。不难发现一条最短路径是由任意路径加上若干环构成的，所以我们只需要维护一个生成森林的信息加上环权值构成的线性基就行了。首先我们需要一个并查集维护连通性以及路径异或和，注意到由于需要支持回退所以我们不能路径压缩。线性基直接使用长度为 30 的数组维护即可，由于数据量小可以暴力回退。

最终时间复杂度 $O(n \log n)$

Code

阅读 CF938G.cpp

CF 878C

n 个运动员, m 个项目, 第 i 个运动员第 j 个项目的能力是 $s_{i,j}$.
现在对于 n 次询问 $k = 1, 2, \dots, n$, 让前 k 个运动员 $1, \dots, k$ 进行淘汰赛, 淘汰赛共 $k - 1$ 轮每轮可以选择任意两个没有被淘汰的运动员比赛任意一个项目。问最后可能夺冠的有多少人。

Solution

建图, 将每个运动员看成点, 如果一个运动员可以战胜另外一个运动员就连一条有向边。注意到一个运动员可能夺冠当且仅当存在一个以它为根的有根树。

Solution

建图, 将每个运动员看成点, 如果一个运动员可以战胜另外一个运动员就连一条有向边。注意到一个运动员可能夺冠当且仅当存在一个以它为根的有根树。

进一步观察, 如果将这个图强连通缩点那么可能夺冠的运动员只能是第一个强连通块内的, 同时如果我们用每个块每个项目内最弱到最强的能力值将每个连通块抽象成一个高维长方体, 那么这些长方体一定是严格偏序的, 也就是说缩完点的 dag 一定是满的

进一步观察, 如果将这个图强连通缩点那么可能夺冠的运动员只能是第一个强连通块内的, 同时如果我们用每个块每个项目内最弱到最强的能力值将每个连通块抽象成一个高维长方体, 那么这些长方体一定是严格偏序的, 也就是说缩完点的 dag 一定是满的
接下来考虑插入, 由于我们已经有了强连通图的顺序结构, 所以可以直接二分新的点会把哪些块粘在一起。

Code

阅读 CF878C.cpp

作业

- 模拟赛补题
- CF 1654E, 1555F, 878C, 938G
- Luogu P5471

Thanks!