题解

whzzt

2019 年 3 月 15 日

安徽师范大学附属中学

Tried

给一个图，求图上能走出的最大数字是多少和方案数。

Tried

给一个图，求图上能走出的最大数字是多少和方案数。

$n \leq 10^5, m \leq 10^6$

Solution

若图中存在一条非 0 边可以到达任意一个环，我们可以直接输出两个 inf，否则我们可以将 0 环给缩成一个点，方案数记为 inf。于是只要考虑图是一个有向无环图时的情况。

Solution

若图中存在一条非 0 边可以到达任意一个环，我们可以直接输出两个 inf，否则我们可以将 0 环给缩成一个点，方案数记为 inf。于是只要考虑图是一个有向无环图时的情况。

接下来我们可以求出每个点向下得到的最大位数，随后逐位确定即可,。

简单的数论题

求

$$\sum_{a=1}^{n}\sum_{b=1}^{n}\sum_{c=1}^{n}\sum_{d=1}^{n} A_a B_b C_c D_d E_{\gcd(a,b)} F_{\gcd(b,c)} G_{\gcd(c,d)} H_{\gcd(d,a)}$$

$n \le 5 \times 10^4$

Solution

将 E, F, G, H 卷上 μ，A, B, C, D 做高维后缀和后，并交换一些序列后，原式可以化为以下形式：

$$\sum_{a=1}^{n}\sum_{b=1}^{n}\sum_{c=1}^{n}\sum_{d=1}^{n} A_a B_b C_c D_d E_{\operatorname{lcm}(a,b)} F_{\operatorname{lcm}(b,c)} G_{\operatorname{lcm}(c,d)} H_{\operatorname{lcm}(d,a)}$$

Solution

将 E, F, G, H 卷上 μ，A, B, C, D 做高维后缀和后，并交换一些序列后，原式可以化为以下形式：

$$\sum_{a=1}^{n}\sum_{b=1}^{n}\sum_{c=1}^{n}\sum_{d=1}^{n} A_a B_b C_c D_d E_{\operatorname{lcm}(a,b)} F_{\operatorname{lcm}(b,c)} G_{\operatorname{lcm}(c,d)} H_{\operatorname{lcm}(d,a)}$$

注意到 $\operatorname{lcm}(a,b) \le n$ 的 (a,b) 对数是 $O(n\log^2 n)$ 的，我们可以将这些点对间连上边，接下来的问题就是一个有关四元环的计数问题。

Solution

将 E, F, G, H 卷上 μ，A, B, C, D 做高维后缀和后，并交换一些序列后，原式可以化为以下形式:

$$\sum_{a=1}^{n}\sum_{b=1}^{n}\sum_{c=1}^{n}\sum_{d=1}^{n} A_a B_b C_c D_d E_{\mathrm{lcm}(a,b)} F_{\mathrm{lcm}(b,c)} G_{\mathrm{lcm}(c,d)} H_{\mathrm{lcm}(d,a)}$$

注意到 $\mathrm{lcm}(a, b) \le n$ 的 (a, b) 对数是 $O(n \log^2 n)$ 的，我们可以将这些点对间连上边，接下来的问题就是一个有关四元环的计数问题。

按照点度从小到大将点排序，那么每个点至多与后面的 $\sqrt{m}$ 个点有边。枚举最靠后的点在四元环上对侧的点，枚举这两点间的路径统计即可。

Solution

将 E, F, G, H 卷上 μ，A, B, C, D 做高维后缀和后，并交换一些序列后，原式可以化为以下形式：

$$\sum_{a=1}^{n}\sum_{b=1}^{n}\sum_{c=1}^{n}\sum_{d=1}^{n} A_a B_b C_c D_d E_{\mathrm{lcm}(a,b)} F_{\mathrm{lcm}(b,c)} G_{\mathrm{lcm}(c,d)} H_{\mathrm{lcm}(d,a)}$$

注意到 $\mathrm{lcm}(a, b) \le n$ 的 (a, b) 对数是 $O(n \log^2 n)$ 的，我们可以将这些点对间连上边，接下来的问题就是一个有关四元环的计数问题。

按照点度从小到大将点排序，那么每个点至多与后面的 $\sqrt{m}$ 个点有边。枚举最靠后的点在四元环上对侧的点，枚举这两点间的路径统计即可。

时间复杂度 $O(m\sqrt{m})$。

恶熊咆哮

平面上有 n 只熊 (x_i, y_i)，熊会按顺序叫，当一只熊叫时，所有熊都会选择周围的 3×3 区域中离目标最近的格子走过去。熊会按照编号从小到大叫，对所有 i 求若第 i 只熊不会叫也不会动，最终所有熊的位置的横纵坐标之积的和。

恶熊咆哮

平面上有 n 只熊 (x_i, y_i)，熊会按顺序叫，当一只熊叫时，所有熊都会选择周围的 3×3 区域中离目标最近的格子走过去。熊会按照编号从小到大叫，对所有 i 求若第 i 只熊不会叫也不会动，最终所有熊的位置的横纵坐标之积的和。

$n \leq 2 \times 10^5$

Solution

容易发现对于熊的移动来说，行列是独立的，接下来我们只考虑一维的情况。考虑一只熊叫意味着什么，我们将所有熊的位置排好序并差分，那么一只熊叫就意味着将其左边的第一个不为 0 的值减去 1，并将其右边的第一个不为 0 的值减去 1。

Solution

容易发现对于熊的移动来说，行列是独立的，接下来我们只考虑一维的情况。考虑一只熊叫意味着什么，我们将所有熊的位置排好序并差分，那么一只熊叫就意味着将其左边的第一个不为 0 的值减去 1，并将其右边的第一个不为 0 的值减去 1。

接下来我们不考虑病熊，模拟以上过程。当一只熊叫的时候，它会将 i 位置和 j 位置的值减去 1，于是我们可以在这两个位置打上标记。接下来我们尝试维护这样的标记，使得第 i 只熊病了恰好会产生在 L_i 和 R_i 位置加 1 的效果，这样我们就能维护答案了。

Solution

考虑一只熊吼叫时对标记的影响，对于一堆标记 L 和 R 来说，显然会产生下面的影响：

Solution

考虑一只熊吼叫时对标记的影响，对于一堆标记 L 和 R 来说，显然会产生下面的影响：

1. 若 L 和 R 均在当前两侧的 0 构成的区间内，且 L 和 R 在吼叫的熊的两侧，或是 L 和 R 中有一个在区间内的情况下，标记会被 push 到该侧第一个不为 0 的位置。

Solution

考虑一只熊吼叫时对标记的影响，对于一堆标记 L 和 R 来说，显然会产生下面的影响：

1. 若 L 和 R 均在当前两侧的 0 构成的区间内，且 L 和 R 在吼叫的熊的两侧，或是 L 和 R 中有一个在区间内的情况下，标记会被 push 到该侧第一个不为 0 的位置。

2. 当 L 和 R 均在区间内且在左侧时，R 会到达 L 的位置，L 会被 push 走，同在右侧也是类似的。

Solution

考虑一只熊吼叫时对标记的影响，对于一堆标记 L 和 R 来说，显然会产生下面的影响：

1. 若 L 和 R 均在当前两侧的 0 构成的区间内，且 L 和 R 在吼叫的熊的两侧，或是 L 和 R 中有一个在区间内的情况下，标记会被 push 到该侧第一个不为 0 的位置。

2. 当 L 和 R 均在区间内且在左侧时，R 会到达 L 的位置，L 会被 push 走，同在右侧也是类似的。

如果暴力处理这些 pair 的话，时间复杂度是 $O(n^2)$ 的。

Solution

考虑对于一对 L 和 R 来说，它们之间的数一定全都是 0，因此所有影响到的 pair 的 L 和 R 一定都在区间内。

Solution

考虑对于一对 L 和 R 来说，它们之间的数一定全都是 0，因此所有影响到的 pair 的 L 和 R 一定都在区间内。

考虑对于一个 pair (L, R) 来说，我们在 R 位置的 left 中插入 L，或是在 L 位置的 right 中插入 R，接下来考虑维护这样的 left 和 right。

Solution

考虑对于一对 L 和 R 来说，它们之间的数一定全都是 0，因此所有影响到的 pair 的 L 和 R 一定都在区间内。

考虑对于一个 pair (L,R) 来说，我们在 R 位置的 left 中插入 L，或是在 L 位置的 right 中插入 R，接下来考虑维护这样的 left 和 right。

对于当前点左侧的 left，它们显然可以被并入左端点的 right 中；
对于当前点右侧的 right，它们显然可以被并入右端点的 left 中；

Solution

考虑对于一对 L 和 R 来说，它们之间的数一定全都是 0，因此所有影响到的 pair 的 L 和 R 一定都在区间内。

考虑对于一个 pair (L, R) 来说，我们在 R 位置的 left 中插入 L，或是在 L 位置的 right 中插入 R，接下来考虑维护这样的 left 和 right。

对于当前点左侧的 left，它们显然可以被并入左端点的 right 中；对于当前点右侧的 right，它们显然可以被并入右端点的 left 中；

对于当前点左侧的 right，我们分成 right 到达的点在两侧来处理，容易发现同样可以并入左端点的 right 中；另一侧的情况也是类似的。

Solution

考虑对于一对 L 和 R 来说，它们之间的数一定全都是 0，因此所有影响到的 pair 的 L 和 R 一定都在区间内。

考虑对于一个 pair (L, R) 来说，我们在 R 位置的 left 中插入 L，或是在 L 位置的 right 中插入 R，接下来考虑维护这样的 left 和 right。

对于当前点左侧的 left，它们显然可以被并入左端点的 right 中；对于当前点右侧的 right，它们显然可以被并入右端点的 left 中；

对于当前点左侧的 right，我们分成 right 到达的点在两侧来处理，容易发现同样可以并入左端点的 right 中；另一侧的情况也是类似的。

容易证明，启发式合并的复杂度为 $O(n \log n)$。