数据结构

mcfx

2019 年 7 月 3 日

Strange Road System

$N(N \leq 10^5)$ 个点，支持添一条无向边，和求 x 到 y 的异或值最大的路径的异或值。

HE Strange Road System

Strange Road System

$N(N \leq 10^5)$ 个点，支持添一条无向边，和求 x 到 y 的异或值最大的路径的异或值。

用带权并查集维护当前的一棵生成树中，每个点到根的路径的异或值。对于非树边，维护环的异或的线性基。

Strange Road System

$N(N \leq 10^5)$ 个点，支持添一条无向边，和求 x 到 y 的异或值最大的路径的异或值。

用带权并查集维护当前的一棵生成树中，每个点到根的路径的异或值。对于非树边，维护环的异或的线性基。
合并两个集合时同时需要合并线性基。复杂度 $O(M \log^2 N)$。

Colorblind Feast

维护一个序列，支持在前面或后面添加 (a_i, b_i)，以及给出一个 l，询问所有区间中，满足 $l \le a_i \le l + k$ 的 b_i 之和的最大值。
$N, M \le 10^5$, b_i 可能有负数。

CC CBFEAST

Colorblind Feast

维护一个序列，支持在前面或后面添加 (a_i, b_i)，以及给出一个 l，询问所有区间中，满足 $l \le a_i \le l + k$ 的 b_i 之和的最大值。
$N, M \le 10^5$, b_i 可能有负数。

$l \le a_i \le r$ 的最大子段和。最大子段和可以用一般的方法维护。

Colorblind Feast

维护一个序列，支持在前面或后面添加 (a_i, b_i)，以及给出一个 l，询问所有区间中，满足 $l \leq a_i \leq l + k$ 的 b_i 之和的最大值。
$N, M \leq 10^5$, b_i 可能有负数。

$l \leq a_i \leq r$ 的最大子段和。最大子段和可以用一般的方法维护。
维护两个值域线段树，一个表示在前面插入，一个表示在后面插入。每次添加操作在对应的线段树区间打标记。

树上 GCD

给一棵有根树，定义
$f(u,v)=gcd(dis(u,lca(u,v)),dis(v,lca(u,v)))$，问对于 $1\sim n$ 中每个 i，有多少个 $f(u,v)=i$。
$N\leq 10^5$。

树上 GCD

给一棵有根树，定义 $f(u,v)=gcd(dis(u,lca(u,v)),dis(v,lca(u,v)))$，问对于 $1\sim n$ 中每个 i，有多少个 $f(u,v)=i$。
$N\le 10^5$。

先将问题转化为求有多少个 $f(u,v)$ 是 i 的倍数。

UOJ 33

树上 GCD

给一棵有根树，定义
$f(u,v) = gcd(dis(u, lca(u,v)), dis(v, lca(u,v)))$，问对于 $1 \sim n$ 中每个 i，有多少个 $f(u,v) = i$。
$N \leq 10^5$。

先将问题转化为求有多少个 $f(u,v)$ 是 i 的倍数。
考虑长链剖分，合并时枚举短链上的长度作为 i，那么需要快速求出链上有多少个值 $\mod i = 0$。

UOJ 33

树上 GCD

给一棵有根树，定义
$f(u,v) = gcd(dis(u, lca(u,v)), dis(v, lca(u,v)))$，问对于 $1 \sim n$ 中每个 i，有多少个 $f(u,v) = i$。
$N \le 10^5$。

先将问题转化为求有多少个 $f(u,v)$ 是 i 的倍数。
考虑长链剖分，合并时枚举短链上的长度作为 i，那么需要快速求出链上有多少个值 $\mod i = 0$。
这个可以对 i 根号分治，小于根号的直接记下 cnt，大于根号的暴力。

CF 878 D

Magic Breeding

初始给 k 个长为 n 的序列，有 m 种操作。
1. 两个序列对应位置取 min，得到一个新序列，编号为当前序列个数 $+1$。
2. 两个序列对应位置取 max，得到一个新序列，编号为当前序列个数 $+1$。
3. 求 x 序列的 y 位置的值。
$n, m \leq 1e5, k \leq 12$。

CF 878 D

Magic Breeding

初始给 k 个长为 n 的序列，有 m 种操作。
1. 两个序列对应位置取 min，得到一个新序列，编号为当前序列个数 $+1$。
2. 两个序列对应位置取 max，得到一个新序列，编号为当前序列个数 $+1$。
3. 求 x 序列的 y 位置的值。
$n, m \leq 1e5, k \leq 12$。

如果是 01 序列，那么取 min 相当于 and，取 max 相当于 or。

CF 878 D

Magic Breeding

初始给 k 个长为 n 的序列，有 m 种操作。
1. 两个序列对应位置取 min，得到一个新序列，编号为当前序列个数 $+1$。
2. 两个序列对应位置取 max，得到一个新序列，编号为当前序列个数 $+1$。
3. 求 x 序列的 y 位置的值。
$n, m \le 1e5, k \le 12$。

如果是 01 序列，那么取 min 相当于 and，取 max 相当于 or。
对每个序列可以维护长为 2^k 的 bitset，第 i 个位置表示某个值在初始 k 个序列的状态是 i 时，在当前序列中的值。

Magic Breeding

初始给 k 个长为 n 的序列，有 m 种操作。
1. 两个序列对应位置取 min，得到一个新序列，编号为当前序列个数 $+1$。
2. 两个序列对应位置取 max，得到一个新序列，编号为当前序列个数 $+1$。
3. 求 x 序列的 y 位置的值。
$n, m \leq 1e5, k \leq 12$。

如果是 01 序列，那么取 min 相当于 and，取 max 相当于 or。
对每个序列可以维护长为 2^k 的 bitset，第 i 个位置表示某个值在初始 k 个序列的状态是 i 时，在当前序列中的值。
回答询问时直接求出 k 个序列中的状态即可。

CF 878 D

Magic Breeding

初始给 k 个长为 n 的序列，有 m 种操作。
1. 两个序列对应位置取 min，得到一个新序列，编号为当前序列个数 $+1$。
2. 两个序列对应位置取 max，得到一个新序列，编号为当前序列个数 $+1$。
3. 求 x 序列的 y 位置的值。
$n, m \le 1e5, k \le 12$。

如果是 01 序列，那么取 min 相当于 and，取 max 相当于 or。
对每个序列可以维护长为 2^k 的 bitset，第 i 个位置表示某个值在初始 k 个序列的状态是 i 时，在当前序列中的值。
回答询问时直接求出 k 个序列中的状态即可。
如果不是 01 序列，那么可以二分，$\ge mid$ 的变成 1，$< mid$ 的变成 0。然后再套用上面的做法就行了。

Magic Breeding

初始给 k 个长为 n 的序列，有 m 种操作。
1. 两个序列对应位置取 min，得到一个新序列，编号为当前序列个数 $+1$。
2. 两个序列对应位置取 max，得到一个新序列，编号为当前序列个数 $+1$。
3. 求 x 序列的 y 位置的值。
$n, m \leq 1e5, k \leq 12$。

如果是 01 序列，那么取 min 相当于 and，取 max 相当于 or。
对每个序列可以维护长为 2^k 的 bitset，第 i 个位置表示某个值在初始 k 个序列的状态是 i 时，在当前序列中的值。
回答询问时直接求出 k 个序列中的状态即可。
如果不是 01 序列，那么可以二分，$\geq mid$ 的变成 1，$< mid$ 的变成 0。然后再套用上面的做法就行了。
由于只需要对 k 个位置二分，复杂度为 $O(nk + m \cdot \frac{2^k}{w} + m \log k)$。

UOJ 218

火车管理

区间入栈，单点弹栈，求区间栈顶和。$N, M \leq 1e5$。

UOJ 218

火车管理

区间入栈，单点弹栈，求区间栈顶和。$N, M \leq 1e5$。

一棵主席树维护每个位置的入栈时间，一棵线段树维护区间和。

UOJ 218

火车管理

区间入栈，单点弹栈，求区间栈顶和。$N, M \leq 1e5$。

一棵主席树维护每个位置的入栈时间，一棵线段树维护区间和。
入栈时将入栈时间修改为 i，然后更新一下值。

UOJ 218

火车管理

区间入栈，单点弹栈，求区间栈顶和。$N, M \le 1e5$。

一棵主席树维护每个位置的入栈时间，一棵线段树维护区间和。
入栈时将入栈时间修改为 i，然后更新一下值。
弹栈时在主席树上查询 (当前入栈时间 $-1)$ 时刻的入栈时间。

Unknown

有一个元素为向量的序列 S，初始时为空，你需要支持三个操作：
1. 在 S 的末尾添加一个元素 (x, y)。
2. 删除 S 的末尾元素。
3. 询问下标在 $[l, r]$ 区间内的元素中，$(x, y) \times S_i$ 的最大值。其中 $\times$ 表示向量的叉积。

操作次数 $\leq 10^5$。

UOJ 191

Unknown

有一个元素为向量的序列 S，初始时为空，你需要支持三个操作：
1. 在 S 的末尾添加一个元素 (x, y)。
2. 删除 S 的末尾元素。
3. 询问下标在 $[l, r]$ 区间内的元素中，$(x, y) \times S_i$ 的最大值。其中 $\times$ 表示向量的叉积。

操作次数 $\leq 10^5$。

如果没有删除操作，显然可以用线段树维护凸包。但是有删除操作之后，可以不停添加并删除一个节点，然后复杂度就爆了。

Unknown

有一个元素为向量的序列 S，初始时为空，你需要支持三个操作：
1. 在 S 的末尾添加一个元素 (x, y)。
2. 删除 S 的末尾元素。
3. 询问下标在 $[l, r]$ 区间内的元素中，$(x, y) \times S_i$ 的最大值。其中 $\times$ 表示向量的叉积。

操作次数 $\leq 10^5$。

如果没有删除操作，显然可以用线段树维护凸包。但是有删除操作之后，可以不停添加并删除一个节点，然后复杂度就爆了。
有一种保证复杂度正确的方法：加入操作时，如果一个节点右边的同级节点已经满了，才重构当前节点的凸包。

Unknown

有一个元素为向量的序列 S，初始时为空，你需要支持三个操作：
1. 在 S 的末尾添加一个元素 (x, y)。
2. 删除 S 的末尾元素。
3. 询问下标在 $[l, r]$ 区间内的元素中，$(x, y) \times S_i$ 的最大值。其中 $\times$ 表示向量的叉积。
操作次数 $\leq 10^5$。

如果没有删除操作，显然可以用线段树维护凸包。但是有删除操作之后，可以不停添加并删除一个节点，然后复杂度就爆了。
有一种保证复杂度正确的方法：加入操作时，如果一个节点右边的同级节点已经满了，才重构当前节点的凸包。
那么要重构一个长为 L 的节点，需要 $O(L)$ 个操作。

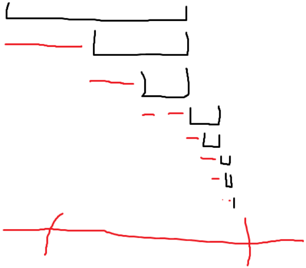
Unknown

有一个元素为向量的序列 S，初始时为空，你需要支持三个操作：
1. 在 S 的末尾添加一个元素 (x, y)。
2. 删除 S 的末尾元素。
3. 询问下标在 $[l, r]$ 区间内的元素中，$(x, y) \times S_i$ 的最大值。其中 $\times$ 表示向量的叉积。

操作次数 $\leq 10^5$。

如果没有删除操作，显然可以用线段树维护凸包。但是有删除操作之后，可以不停添加并删除一个节点，然后复杂度就爆了。
有一种保证复杂度正确的方法：加入操作时，如果一个节点右边的同级节点已经满了，才重构当前节点的凸包。
那么要重构一个长为 L 的节点，需要 $O(L)$ 个操作。
证明在下一页。

考虑已经加满而没有重构的节点，他们下面一层的节点（即所有自身已经重构，而父亲节点没有重构的这些节点）显然是两两不相交，刚好构成了整个序列。
进入这些节点的次数是 $O(\log L)$，在这些节点中，左右端点递归到底层的次数是 $O(\log L)$，中间的个数也是 $O(\log L)$。



HDU 6369

Rectangle Outline

给 $n(n \leq 10^5)$ 个矩形，如果他们并集只有外边界线，那么沿顺时针方向输出外边界线上的所有转折点。

HDU 6369

Rectangle Outline

给 $n(n \leq 10^5)$ 个矩形，如果他们并集只有外边界线，那么沿顺时针方向输出外边界线上的所有转折点。

首先可以考虑求出边界线上的所有转折点。

HDU 6369

Rectangle Outline

给 $n(n \leq 10^5)$ 个矩形，如果他们并集只有外边界线，那么沿顺时针方向输出外边界线上的所有转折点。

首先可以考虑求出边界线上的所有转折点。
可以发现每个点出现当且仅当他上面或下面的左右两边有一侧覆盖次数是 0，而另一侧不是。

HDU 6369

Rectangle Outline

给 $n(n \le 10^5)$ 个矩形，如果他们并集只有外边界线，那么沿顺时针方向输出外边界线上的所有转折点。

首先可以考虑求出边界线上的所有转折点。
可以发现每个点出现当且仅当他上面或下面的左右两边有一侧覆盖次数是 0，而另一侧不是。
那么可以扫描线，维护线段树区间加，把修改前/后是 0 的区间拿出来求出所有交点。

HDU 6369

Rectangle Outline

给 $n(n \le 10^5)$ 个矩形，如果他们并集只有外边界线，那么沿顺时针方向输出外边界线上的所有转折点。

首先可以考虑求出边界线上的所有转折点。
可以发现每个点出现当且仅当他上面或下面的左右两边有一侧覆盖次数是 0，而另一侧不是。
那么可以扫描线，维护线段树区间加，把修改前/后是 0 的区间拿出来求出所有交点。
只有外边界线的条件可以在点数超过 $8n-4$ 时 break 来处理。

HDU 6369

Rectangle Outline

给 $n(n \leq 10^5)$ 个矩形，如果他们并集只有外边界线，那么沿顺时针方向输出外边界线上的所有转折点。

首先可以考虑求出边界线上的所有转折点。
可以发现每个点出现当且仅当他上面或下面的左右两边有一侧覆盖次数是 0，而另一侧不是。
那么可以扫描线，维护线段树区间加，把修改前/后是 0 的区间拿出来求出所有交点。
只有外边界线的条件可以在点数超过 $8n-4$ 时 break 来处理。
最后是输出答案。有一个性质是，横/纵坐标相同的点中，两两相邻的之间的连边一定在边界上，所以直接开一堆 vector，每次跳。

Number Queries

维护一个集合，支持把 $\geq x$ 的 $-= x$ 和求值域在 $[l, r]$ 的第 k 大。

$n, m \leq 10^5, v \leq 10^{18}$

Number Queries

维护一个集合，支持把 $\geq x$ 的 $-=x$ 和求值域在 $[l, r]$ 的第 k 大。
$n, m \leq 10^5, v \leq 10^{18}$

用平衡树维护，2 操作很好实现。

HE Number Queries

Number Queries

维护一个集合，支持把 $\geq x$ 的 $-= x$ 和求值域在 $[l, r]$ 的第 k 大。
$n, m \leq 10^5, v \leq 10^{18}$

用平衡树维护，2 操作很好实现。
对于 1 操作，首先分成两棵树。

HE Number Queries

Number Queries

维护一个集合，支持把 $\geq x$ 的 $-=x$ 和求值域在 $[l,r]$ 的第 k 大。
$n,m \leq 10^5, v \leq 10^{18}$

用平衡树维护，2 操作很好实现。
对于 1 操作，首先分成两棵树。
如果较大树的 $\min \geq$ 较小树的 max，那么可以直接减。

HE Number Queries

Number Queries

维护一个集合，支持把 $\geq x$ 的 $-= x$ 和求值域在 $[l, r]$ 的第 k 大。
$n, m \leq 10^5, v \leq 10^{18}$

用平衡树维护，2 操作很好实现。
对于 1 操作，首先分成两棵树。
如果较大树的 $\min \geq$ 较小树的 max，那么可以直接减。
否则较大树的 $\min$ 至少会减半，可以把这部分的暴力拿出来做。

Naive Operations

给两个序列 a, b，a 初始为全 0，b 是排列，有两种操作：区间 $a_i + 1$，求区间的 $\frac{a_i}{b_i}$ 之和。$n, m \leq 10^5$。

HDU 6315

Naive Operations

给两个序列 a, b，a 初始为全 0，b 是排列，有两种操作：区间 a_i+1，求区间的 $\frac{a_i}{b_i}$ 之和。$n, m \le 10^5$。

用线段树维护每个位置还要多少次能让值 $+1$，区间减时暴力变成 0 的位置。

HDU 6315

Naive Operations

给两个序列 a, b，a 初始为全 0，b 是排列，有两种操作：区间 $a_i + 1$，求区间的 $\frac{a_i}{b_i}$ 之和。$n, m \leq 10^5$。

用线段树维护每个位置还要多少次能让值 $+1$，区间减时暴力变成 0 的位置。
复杂度 $O(m \log^2 n)$。

Road Trip

有 n 个城市，依次以道路相连。i 到 $i+1$ 的距离是 w_i。
如果你在 i，你可以免费得到 g_i 升的汽油。此外也可以花 p_i 每升买汽油。
每公里需要一升汽油，容量是无限的。
q 次询问从 x 到 y 需要花多少钱。
$n, q \leq 2 \cdot 10^5, x \leq y$

HR Road Trip

Road Trip

有 n 个城市，依次以道路相连。i 到 $i+1$ 的距离是 w_i。
如果你在 i，你可以免费得到 g_i 升的汽油。此外也可以花 p_i 每升买汽油。
每公里需要一升汽油，容量是无限的。
q 次询问从 x 到 y 需要花多少钱。
$n, q \leq 2 \cdot 10^5, x \leq y$

每当油不够走这一段时，可以在前面最便宜的地方买好用。

Road Trip

有 n 个城市，依次以道路相连。i 到 $i+1$ 的距离是 w_i。
如果你在 i，你可以免费得到 g_i 升的汽油。此外也可以花 p_i 每升买汽油。
每公里需要一升汽油，容量是无限的。
q 次询问从 x 到 y 需要花多少钱。
$n, q \leq 2 \cdot 10^5, x \leq y$

每当油不够走这一段时，可以在前面最便宜的地方买好用。
单调栈可以预处理出 $next_i$，表示从 i 开始，最远能到的地方。

HR Road Trip

Road Trip

有 n 个城市，依次以道路相连。i 到 $i+1$ 的距离是 w_i。
如果你在 i，你可以免费得到 g_i 升的汽油。此外也可以花 p_i 每升买汽油。
每公里需要一升汽油，容量是无限的。
q 次询问从 x 到 y 需要花多少钱。
$n, q \le 2 \cdot 10^5, x \le y$

每当油不够走这一段时，可以在前面最便宜的地方买好用。
单调栈可以预处理出 $next_i$，表示从 i 开始，最远能到的地方。
如果建一棵树，$n+1$ 是根，i 到 $next_i+1$ 连边，那么可以从询问的 x 暴力往上跳。

Road Trip

有 n 个城市，依次以道路相连。i 到 $i+1$ 的距离是 w_i。
如果你在 i，你可以免费得到 g_i 升的汽油。此外也可以花 p_i 每升买汽油。
每公里需要一升汽油，容量是无限的。
q 次询问从 x 到 y 需要花多少钱。
$n, q \le 2 \cdot 10^5, x \le y$

每当油不够走这一段时，可以在前面最便宜的地方买好用。
单调栈可以预处理出 $next_i$，表示从 i 开始，最远能到的地方。
如果建一棵树，$n+1$ 是根，i 到 $next_i + 1$ 连边，那么可以从询问的 x 暴力往上跳。
这棵树有一个特性，$i+1$ 到 i 的路径，一定是若干条返祖边和最后去 i 的一条边。

HR Road Trip

Road Trip

有 n 个城市，依次以道路相连。i 到 $i+1$ 的距离是 w_i。
如果你在 i，你可以免费得到 g_i 升的汽油。此外也可以花 p_i 每升买汽油。
每公里需要一升汽油，容量是无限的。
q 次询问从 x 到 y 需要花多少钱。
$n, q \le 2 \cdot 10^5, x \le y$

每当油不够走这一段时，可以在前面最便宜的地方买好用。
单调栈可以预处理出 $next_i$，表示从 i 开始，最远能到的地方。
如果建一棵树，$n+1$ 是根，i 到 $next_i + 1$ 连边，那么可以从询问的 x 暴力往上跳。
这棵树有一个特性，$i+1$ 到 i 的路径，一定是若干条返祖边和最后去 i 的一条边。
那么可以以 $O(n)$ 代价维护出当前需要额外油的位置。

HR Road Trip

Road Trip

有 n 个城市，依次以道路相连。i 到 $i+1$ 的距离是 w_i。
如果你在 i，你可以免费得到 g_i 升的汽油。此外也可以花 p_i 每升买汽油。
每公里需要一升汽油，容量是无限的。
q 次询问从 x 到 y 需要花多少钱。
$n, q \le 2 \cdot 10^5, x \le y$

每当油不够走这一段时，可以在前面最便宜的地方买好用。
单调栈可以预处理出 $next_i$，表示从 i 开始，最远能到的地方。
如果建一棵树，$n+1$ 是根，i 到 $next_i+1$ 连边，那么可以从询问的 x 暴力往上跳。
这棵树有一个特性，$i+1$ 到 i 的路径，一定是若干条返祖边和最后去 i 的一条边。
那么可以以 $O(n)$ 代价维护出当前需要额外油的位置。
用一棵线段树和另一个单调栈可以维护每个位置加油的代价。

Road Trip

有 n 个城市，依次以道路相连。i 到 $i+1$ 的距离是 w_i。
如果你在 i，你可以免费得到 g_i 升的汽油。此外也可以花 p_i 每升买汽油。
每公里需要一升汽油，容量是无限的。
q 次询问从 x 到 y 需要花多少钱。
$n, q \leq 2 \cdot 10^5, x \leq y$

每当油不够走这一段时，可以在前面最便宜的地方买好用。
单调栈可以预处理出 $next_i$，表示从 i 开始，最远能到的地方。
如果建一棵树，$n+1$ 是根，i 到 $next_i+1$ 连边，那么可以从询问的 x 暴力往上跳。
这棵树有一个特性，$i+1$ 到 i 的路径，一定是若干条返祖边和最后去 i 的一条边。
那么可以以 $O(n)$ 代价维护出当前需要额外油的位置。
用一棵线段树和另一个单调栈可以维护每个位置加油的代价。

Replace

给一个序列，支持区间 x 改成 y，问区间 x 出现次数。
$N \leq 5 \cdot 10^5$。

Replace

给一个序列，支持区间 x 改成 y，问区间 x 出现次数。
$N \leq 5 \cdot 10^5$。

每个值维护一个线段树，然后线段树分裂合并。

CF 799 F

Beautiful fountains rows

有一个 n 行 m 列的网格图，每行有连续一段是染色的，你需要求出有多少对 l, r，满足 $l \le r$ 且每行的 l 到 r 之间的染色的个数是 0 或者奇数，
$n, m \le 10^5$。

CF 799 F

Beautiful fountains rows

有一个 n 行 m 列的网格图，每行有连续一段是染色的，你需要求出有多少对 l,r，满足 $l \leq r$ 且每行的 l 到 r 之间的染色的个数是 0 或者奇数，
$n,m \leq 10^5$。

枚举左右端点的奇偶，每一行给左右端点了若干个限制，每个限制是一个矩形。那么对这些矩形的并求交即可。可以扫描线 $+$ 线段树，线段树维护最大值和最大值的个数。