高级数据结构

h10

2018 年 1 月 11 日

准确的说我也不知道哪些数据结构是基础数据结构哪些是高级数据结构

所以先弄几道有关并查集，RMQ，树状数组，线段树的题热热身吧

bzoj4569

有一个长度为 n 的没有前导零的十进制数，用 s 表示，有 m 个限制条件，每个条件形如 $(l1, r1, l2, r2)$，表示 $s[l1:r1] = s[l2:r2]$

现在给出这些限制条件，问有多少个数满足条件

$n, m \leq 10^5$

bzoj4569

考虑类似ST表的方法，我们把这个区间划分成前 2^k 位和后 2^k 位，那么就变成了这两端 2^k 位分别对应相同

我们开 $\log(n)$ 个并查集，第 k 个记录对应的第 k 层的相同性

处理完所有条件之后，我们从上往下把相同性结果推到下一层去，就可以在总时间 $O((n+m)\log(n)\alpha(n))$ 的复杂度内得到最后的并查集

bzoj2957

有 n 栋楼房在一个二维平面上，第 i 栋楼房可以用线段 $(i,0)$ (i,H_i) 表示

最开始楼房的高度全为 0，m 次操作，每次把第 x_i 栋楼房的高度修改为 y_i，并询问在 $(0,0)$ 处可见楼房数量

$1 \le x_i, n, m \le 10^5, 1 \le y_i \le 10^9$

bzoj2957

显而易见，可见的数字是单调递增的，修改一个数只会对后面的数造成影响。那么考虑线段树中的每一条线段

只有两种情况：

1.最大值小于等于修改的数，那么这个线段的贡献为 0，无需处理；

2.否则将这个线段分成两段，如果左侧的最大值大于修改的数，那么不影响右侧贡献，递归处理左侧；否则就变成了第一种情况，递归左侧

±1 RMQ

最后讲一个奇妙的东西：±1 RMQ
使用条件为数组的任意连续两个数字的差的绝对值为 1
树上求lca就可以使用这个

± 1 RMQ

先分块，每一块的大小设为 $\frac{\log(n)}{2}$，则一共有 $l = \frac{2n}{\log(n)}$ 块

把每一块当成一个数字，做普通RMQ，时间为 $O(l\log(l)) = O(\frac{n}{\log(n)}(\log(n) - \log(\log(n)))) = O(n)$

连续的几块的查询就与普通RMQ一样了，重点是块内查询

考虑到块的大小为 $\frac{log(n)}{2}$，每一个数与上一个数的差只能是 1 或 -1，那么本质不同的块就只有 $2^{\frac{log(n)}{2}} = \sqrt{n}$ 个了

对于一种种块，预处理出所有有可能的块内查询，都只有 $2^{\frac{log(n)}{2}} = \sqrt{n}$ 种，所以块内查询的时间为 $O(\sqrt{n}\sqrt{n}) = O(n)$

常见平衡树

Splay：通过rotate使操作结点提升至根，可用势能分析证明复杂度均摊 $\log(n)$

Treap：玄学随机权重 $\log(n)$

替罪羊：玄学调参 $\log(n)$

bzoj2827

平面上有 n 只鸟，每只都有自己的初始位置与威武值

接下来 t 秒间，每秒都会有一只鸟改变自己的位置

定义一只鸟在某一刻的士气值为此刻与它站在同一位置的所有鸟中最大的威武值，不包括自己

定义一只鸟在某一刻的团结值为此刻与它站在同一位置的鸟的个数，不包括自己

求每只鸟的士气值与团结值的历史最大值

bzoj2827

平衡树裸题，写个哈希表来存坐标

我们需要这样一种数据结构，它需要支持插入，删除，集合chkmax，维护集合大小，集合max

随便找种平衡树咯

雅礼wc2017集训题

一棵树，两种操作

1. *Update* x k 更新操作，对于在 x 的子树中的每一个节点，如果它到 x 的距离为 D，那么将它的权值加上 $fib[k+D]$

2. *Query* x y 询问操作，询问 x 到 y 的简单路径上的所有点的权值之和模 10^9+7

$n, m \leq 10^5, v \leq 10^9$

雅礼wc2017集训题

$$fib[a+b] = fib[a-1] * fib[b] + fib[a] * fib[b+1]$$

所以令每个点的dep为它的深度，对于一次操作，相当于 $fib[dep[i]-1] * fib[x] + fib[dep[i]] * fib[x+1]$

维护每个点 $fib[x]$ 的和以及 $fib[x+1]$ 的和即可，树链剖分

QT IV

给你一颗 n 个点的树，有边权，每个点的颜色可能是黑或者白，初始皆为白色，要求支持以下操作

1. 反转 u 的颜色
2. 询问整棵树中最远的白色点对

$n, m \leq 10^5$

QT IV

LCT 维护虚边信息

定义 $lmax(u)$ 表示这个子splay表示的一段重链最浅的点出发到子树内某个白点结束的最长链长，$rmax(u)$ 则表示从该重链最深的点出发到子树内某个白点结束的最长链长

定义 $max(u)$ 为以 u 为根的 Splay 中的这些点以及它们的虚边带出的子树中，最长的白点点对路径

用 $chain(u)$（要用 multiset）维护 u 连出的虚边指向的节点的 $lmax$ 值的集合

用 $path(u)$ 维护 u 连出的虚边指向的节点的 max 值的集合

bzoj4605

平面上有两种操作：

1.在坐标 (x, y) 处放置一个点，权值为 v

2.询问坐标为 $(x_1, y_1), (x_2, y_2)$ 的矩形区域中，权值第 k 大的点的权值

坐标范围 $n \leq 500000$，询问数 $q \leq 100000$，所有权值 $v \leq 10^9$

强制在线

bzoj4605

值域线段树套kd树，由于动态加点，要使用替罪羊思想对kd树定期重构

$O(q\log(v)(\log^2(q)+\sqrt{q}))$，嘴巴起来真TM简单

bzoj3053

k 维空间内有 n 个点，m 次询问，每次询问与一个点前 10 近的点是哪些

bzoj3053

我们把当前取到的前 10 优解放在大根堆中，然后估价函数比较是不是比堆顶的解优就可以了

Unknown

有一个元素为二维向量的序列要求支持以下三种操作

1.队尾插入一个新向量

2.队尾删除

3.查询区间 $[l, r]$ 内所有向量与 (x, y) 叉积的最大值

$n, m \leq 10^5; x, y \leq 10^9$

Unknown

维护区间凸包即可，两个大小为 a 与 b 凸包合并的复杂度是 $O(a+b)$ 的

Unknown

如果没有删除的话，二进制分组是长这样的：

每次加入一个元素到末尾单独成为一块，如果末尾两个块大小相同，那么合并它们，一直合并到不能合并为止

考虑一个元素所在的块每次合并大小会翻倍，因此一个元素贡献的复杂度是 $O(\log(n))$ ，所以构建的复杂度是 $O(n\log(n))$ 的

考虑每次查询，将查询最多分成 $O(\log(n))$ 个块，再加上二分的复杂度就是 $O(\log^2(n))$

如果有删除的话，单次重构的复杂度可能到 $O(n)$ ，总复杂度就是 $O(n^2)$ 了，所以我们需要改进这个做法

Unknown

考虑删除，引入替罪羊思想，对于要重构的组，我们打个标记，并不立马重构，但也不能一直不重构，不然询问的时候访问到的节点数会爆炸。

我们约定：每一层至多允许最后一个组是有标记的

这样的话，当第 i 层需要增加一个组的时候才会将原来的最后一组重构，当这个组被标记完了，那么就直接变成空的即可

可以证明修改与操作的复杂度依然是 $O(n\log(n))$ 的，加上本题询问的二分答案，总复杂度为 $O(n\log^2(n))$

区间第k大

给定一个长度为 n 的序列，要求支持询问区间 $[l, r]$ 中第 k 大的元素的值

主席树裸题

区间第k大

如果单点修改（插入删除）呢？

替罪羊树或Treap套值域线段树，时间 $O(n\log^2(n))$，空间 $O(n\log^2(n))$

值域线段树套线段树，时间 $O(n\log^2(n))$，空间 $O(n\log^2(n))$

值域线段树套平衡树，时间 $O(n\log^2(n))$，空间 $O(n\log(n))$

数据结构

给定 n 个矩形，m 次操作每次询问一个点被多少矩形包含，强制在线

$n, m \leq 10^5; x, y \leq 10^9$

数据结构

考虑离线，那么可以使用扫描线从左到右扫

每次遇到一个矩形则在上端点加一，下端的减一；每次离开一个矩形则在上端点减一，上端的减一

考虑询问的话，求前缀和即可

强制在线的话把扫描线上的内容可持久化一下就好了

谈笑风生

给出一棵 n 个点的树，以及 m 个询问，每个询问形式如下：

给定 p, k，问有多少个三元组 (p, x, y)，满足

1. p 和 x 都是 y 的祖先
2. p 和 x 的距离小于等于 k

$n, m \leq 10^5$

谈笑风生

分两种情况

1. x 是 p 的祖先：非常好处理

2. p 是 x 的祖先或 $x=p$：

此时合法的 x 的深度区间为 $[dep(p), dep[p]+k]$，满足条件的 x 对答案有 $sz[x]-1$ 的贡献

按 DFS 序建立主席树即可

basic persistent treap

初始给定一个长为 n 的序列，有 m 次操作，分别有四种操作

1. t, l, r, k 将 t 时刻的序列区间 $[l, r]$ 加上 k 并将其作为新的序列
2. t, l, r 将 t 时刻的序列区间 $[l, r]$ 翻转并将其作为新的序列
3. t, l, r 询问 t 时刻序列区间 $[l, r]$ 的最大值
4. t, l, r 询问 t 时刻序列区间 $[l, r]$ 的最大值的前驱的值

$n, m \leq 10^5$

basic persistent treap

可持久化Treap裸题

区间 chkmax

对于一个长度为 n 的数组，要求支持以下操作

1. 区间加减
2. 区间 chkmax
3. 求区间最小值
4. 求区间和

$n, m \leq 10^5$

区间 chkmax

把一个区间分成两部分：所有值等于区间最小值分为第一部分；去除所有区间最小值分为第二部分

记录第一部分最小值为 $mn[x]$，第二部分最小值为 $se[x]$

chkmax(v) 时，如果 $v \leq mn[x]$，则直接 return

如果 $v < se[x]$，则相当于对第一部分做加法

如果 $v \geq se[x]$，则递归左右儿子

可以证明时间复杂度是 $O(n\log(n))$ 的

雅礼NOI2017集训题

对于一个长度为 n 的数组，要求支持以下操作

1. 区间加减
2. 区间整除
3. 求区间最小值
4. 求区间和

$n, m \leq 10^5$

雅礼NOI2017集训题

考虑区间的 min 和 max，当 $min - \lfloor \frac{min}{d} \rfloor = max - \lfloor \frac{max}{d} \rfloor$ 的时候，变成区间加

否则递归

历史最值问题

另 A 为原数组，对 A 经行区间加减操作，B 为 A 的历史最大值/最小值/版本和

区间历史最大值之和

设一个 $C_i = B_i - A_i$，如果 $A_i + x > B_i$，则 $C_i = 0$，否则 $C_i = C_i - x$，换句话说 $C_i = max(C_i - x, 0)$

区间历史最小值之和

设一个 $C_i = A_i - B_i$，如果 $A_i + x < B_i$，则 $C_i = 0$，否则 $C_i = C_i + x$，换句话说 $C_i = max(C_i + x, 0)$

区间历史版本和之和

设一个 $C_i = B_i - T * A_i$，那么 $newC_i = (B_i + A_i + x) - (T + 1) * (A_i + x) = C_i - T * x$

NOI2017Day1T1

有一个整数 x，一开始为 0

接下来有 n 个操作，都是以下两种操作中的一种

1. ab 将 x 加上 $a * 2^b$ 其中 a 为整数，b 为自然数

2. k 询问 x 二进制第 k 为的值

$n \leq 10^6, a \leq 10^9, b \leq 30n$

NOI2017Day1T1

令 x_i 表示 x 的第 i 位

对于每一位我们需要维护不低于这一位的第一个 0 与第一个 1

假设我们使用 bitset 压位，定义 $f[i]$ 表示 x_{32i} 至 x_{32i+31} 是否全是 0，$g[i]$ 表示 x_{32i} 至 x_{32i+31} 是否全是 1，那么相对与暴力，bitset 可以把常数缩小 32 倍

假设我们对 f 与 g 使用 bitset 压位，定义 $F[i]$ 表示 $f[32i]$ 至 $f[32i+31]$ 是否全是 0，$G[i]$ 表示 $g[32i]$ 至 $g[32i+31]$ 是否全是 1，那么常数又可以缩小 32 倍

不停这么干下去，最后复杂度为 $O(n*\log(a)*\log_{32}(30n))$，相比与线段树的 $O(n*\log(a)\log_2(30n))$，32叉线段树要王逸松化不少

划分树

见划分树.pdf