Solution

by tsx

star

- 给定一棵 n 个点的树，求出满足以下条件的点集 S 的个数。
  - S 的导出子图中恰有一个点 u 满足 $\deg_u \geq 3$，其中 $\deg_u$ 表示 u 的度数。
- 对 998244353 取模。
- $1 \leq n \leq 5 \times 10^5$

star 题解

- 考虑枚举那个度数 ≥ 3 的点 u，那么只用考虑叶子即可唯一确定该树。
- 若把它当作根，那么每个子树里至多选一个点，且至少要在三个子树中选点。
- 先不考虑后一个条件，那么答案即为 $\prod_v (sz_v + 1)$。
- 那我们需要减掉只选0个，1个和2个点的方案数。
- 0个和1个很好处理。2个的话需要记录一个变量 sum 表示之前遍历过的子树的大小之和，每次到一个新的儿子 v 时只需将 $sum \times sz_v$ 贡献至答案，再更新 sum 即可。
- 直接做是 $O(n^2)$ 的，但容易发现我们其实并不需要真正换根，只需知道 u 的子树外的大小就行了，那它自然是 $n - sz_u$。
- 时间复杂度 $O(n)$。

fenwick

- 维护一个长度为 n 的整数序列 $a_1, a_2, \ldots, a_n$，初始均为 0，执行 m 次操作，有如下两种操作：
  - `1 l r v`：对于每个 $i \in [l, r]$，将序列 a 中区间 $[i - f(i) + 1, i]$ 里的数增加 v。
  - `2 l r`：询问序列 a 中区间 $[l, r]$ 的和。
- 其中 $f(i)$ 定义为 `lowbit(i)`。
- $1 \le n, m \le 10^6$，$1 \le v \le 10^4$

fenwick 题解

- 考虑使用线段树维护，首先将 n 扩展至 2^k 使树状数组结点与线段树的结点一致。
- 记录两个标记 $lazy_{0,i}, lazy_{1,i}$ 分别为区间加以及区间树状数组加的标记。这两个标记可以下传。
- 对于一次修改，如果当前递归到的区间被完全包含，直接打一个区间树状数组加的标记并返回，此时对区间和的贡献容易预处理。
- 否则，若当前递归到的区间与修改区间相交，并且它是树状数组上的一个区间，那么我们打上一个区间加的标记。
- 询问的时候直接查询区间和即可。
- 时间复杂度 $O(m\log n)$。
- 本题也可以使用树状数组维护，常数会更小一些。

ktree

- 小 T 有一个深度为 n 的满 k 叉树，树上每个结点都有一个权值。树上的叶节点从左到右编号为 $1 \sim k^n$。
- 叶节点的权值为 0 或 1，而非叶结点的权值定义为它子树内叶结点的权值总和。
- 开始时叶节点的权值均为 0，小 T 每次操作可以将其中一个叶节点的权值从 0 变为 1。请问有多少长为 k^n 的操作序列满足以下条件：
  - 每次操作结束后，所有非叶结点的儿子的权值的极差不超过 1。
  - 有 m 条限制，第 i 条限制形如 t_i, x_i，表示第 t_i 次操作的叶节点必须为 x_i。
- 对 998244353 取模。
- T 组数据， $1 \le T \le 2 \times 10^5$， $\sum m \le 2 \times 10^5$，$2 \le k \le 10^6$，$1 \le x_i, t_i \le k^n \le 10^{18}$，$x_i$ 互不相同，t_i 互不相同。

ktree 题解

- 考虑前 k^{n-1} 次操作，注意到它们操作完后每个深度为 $n-1$ 的点的权值必然为1，不仅如此，每经过 k^{n-1} 次操作，它们的权值都会增加1。
- 这可以让我们想到将其分治为 k 个规模为 $n-1$ 的子问题，将深度为 $n-1$ 的点看作叶子即可。
- 先来考虑 $m=0$ 的情形。
- 此时我们需要对每个深度为 $n-1$ 的点，确定它每次增加的时候真正增加权值的叶子的编号。由于它有 k 个儿子，方案数自然是 $k!$。
- 设 $f(n,k)$ 为答案，那么有 $f(n,k)=(k!)^{k^{n-1}}\times f(n-1,k)$，也可以得到其封闭形式 $f(n,k)=(k!)^{nk^{n-1}}$。

ktree 题解

- $m \neq 0$ 时怎么做呢？
- 只需考虑如何对每个深度为 $n-1$ 的点分配编号。
- 统计这些点的儿子中有多少个是有限制的，若某个点的儿子中的限制数为 i，那方案数为 $(k-i)!$。
- 接着分治时直接将限制分治下去即可。注意分治后可能会出现 x_i 相同的情况，需要特判无解。
- 开始时将限制按照 t_i 排序，统计时使用哈希表即可做到时间复杂度 $O(k + T\log p + n\sum m)$。

bracket

- 给定一个长为 $2n$ 的括号序列，其中出现了 n 种括号，每种括号恰出现两次（左括号和右括号各一次）。
- 你需要将其划分成两个子序列，使得这两个子序列均为合法的括号序列。
- 输出一组方案，或报告无解。
- T 组数据，$T \leq 2 \times 10^4$，$1 \leq n \leq 10^6$，$\sum n \leq 1.5 \times 10^6$

bracket 题解

- 考虑将每种括号看作一个区间，那么若两个区间相交但互不包含，则它们一定不能在同一个子序列中。
- 将这样的区间之间连一条边，那么有解当且仅当这是一张二分图，选出两个子序列就等价于给这张图二分图染色。
- 直接实现时间复杂度为 $O(n^2)$。
- 考虑 $[l_1,r_1],[l_2,r_2]$ 两个区间，那么它们之间有边的条件为 $l_1<l_2<r_1<r_2$ 或 $l_2<l_1<r_2<r_1$，这是二维数点，且有特殊性（某一维是一段前缀），可以使用主席树优化建图，时空复杂度均为 $O(n\log n)$。
- 但上面这个东西的时间/空间常数实在太大，不太能跑过。

bracket 题解

- 注意到图上的连通块一定是一个原序列上的区间。
- 从左往右考虑每一个括号，用栈维护当前的每一个连通块，对于每一个连通块，用两条链表维护连通块里颜色为1和颜色为2的点集，且按左括号所在位置从左到右排序。
- 如果一个括号被匹配了，那么它就不再会有影响了，于是我们的链表只维护目前还未匹配的左括号。
- 如果新加入的括号是左括号，只需将其本身压入栈即可。
- 否则，需要一直弹栈直到找到匹配的左括号，每次弹栈时，若该连通块的两个链表均非空，则无解，否则，将新加入的括号与链头连边，表示它们颜色不能相同。
- 最后将栈顶的链表更新一下，具体来说，先将链表尾的那个匹配的括号删除，在将另外一个链表与之前弹出的连通块的链表接起来，将连通块合并。
- 输出方案在图上跑二分图染色即可，时间复杂度 $O(\sum n)$。