1.

给一个有根树（1是根），对所有的点x, 定义d数组，d(i)表示x的后代到x的距离是i的点的个数（距离定义为简单路径上边的条数）。

对于所有点的所有d数组，找到第一个出现的最大值的idx。即第一关键字最大，第二关键字最左（i最小，即最近）

数据规模：N = 1e6

io复杂度: 2 \* (N - 1), output: N

这个题是我做的第一个dsu on tree, 比赛的时候听了一堆人的讲解QAQ，自己真是菜的不行的。还是贴了io挂开底层优化强行冲过去的（现在还是那个题的submission里跑的最慢的）。

思路(by cst)：先进行正常的树剖，最后合并链的时候，先合并短的并清空map，长链不清空。

这个可以直接用一个map + max\_id来维护，或者vector + max\_id。

然后比赛的时候大概短路了，暴力合并写了线段树，我树你🐎呢。。。

2. 给一个树，他每个点有个颜色C，定义i的主要颜色集合为以i为根的子树中出现次数最多的1或多个颜色，计算这个每个点对应的这个集合的元素和。

思路 ：

除了要记住别随便拷贝大的集合之外，没什么需要新掌握的。就直接启发式暴力。