题解

整除

特判 $x=1$。当 $x>1$ 时可以同乘 $x-1$ 转化为有多少 x_0 满足 $f(x_0)\cdot(x_0-1) \bmod (x_0^m-1)=0$。

考虑对于一个 x_0 如何判断是否整除。设 $f(x)\cdot(x-1) \bmod (x^m-1)=\sum_{i=0}^{m-1} a_i x^i$。

不断执行以下操作：

- 若 $a_i \ge x_0$，则将 a_i 减去 x_0，将 $a_{(i+1) \bmod m}$ 加上 1。
- 若 $a_i \le -x_0$，则将 a_i 加上 x_0，将 $a_{(i+1) \bmod m}$ 减去 1。

若已经没有满足条件的位置，即 $|a_i|<x_0$，容易发现 x_0 满足条件当且仅当所有 a_i 都等于 $-x_0+1$, 都等于 0 或都等于 x_0-1。

因此可以得到：

- 若初始时所有 a_i 都为 0，则有无穷个正整数满足条件。
- 否则，只需要考虑不超过 $\max\{|a_i|\}+1$ 的正整数，其数量是 $O(n)$ 的。每次操作会使得 a_i 的绝对值之和减少至少 x_0-1，而初始值为 $O(n)$，因此所有 x_0 的总操作次数是 $O(n\log n)$。

用 `std::map` 维护并支持回退即可。时间复杂度 $O(n\log^2 n)$。

词典

设答案为 $D(n)$，$F(n)=\sum_{j=1}^{n}\lfloor 1+\log_2 j\rfloor$。则有

$$
\begin{gathered}
D(1)=1\\
D(n)=F(n)+\min_{1\le k\le n-1}\{[k>1]F(k)+D(k)+D(n-k)\} \quad (n>1)
\end{gathered}
$$

显然 $F(n)$ 是凸函数，因此容易证明 $D(n)$ 也是凸函数。容易得到一个 $O(n\log n)$ 或 $O(n)$ 的做法。

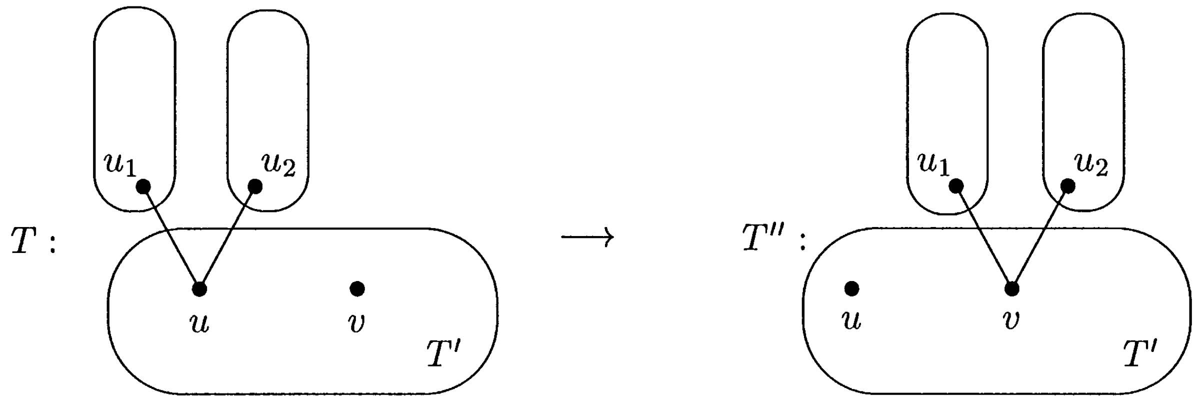
容易发现 $F(n)$ 是 $O(n\log n)$ 级别，因此 $D(n)$ 是 $O(n\log^2 n)$ 级别，因此 $D(n)-D(n-1)$ 是 $O(\log^2 n)$ 级别，可以考虑对每个 v 求出最大的 m 使得 $D(m)-D(m-1)=v$，查询时二分即可。

求 m 可以使用倍增 + 二分，注意倍增时无法每次乘 2，因为求值时 $D(n-k)$ 可能还没有求出。一种方法是当 n 小时暴力，n 足够大时每次乘 1.5，求 $D(n)$ 时三分的下界设为 $n/3$，这样就能避免询问未求出的值。

时间复杂度 $O(\log^5 n)$。也可以选择本地打表。

蒲公英

对于一个以 u 为根的已经编号好的树 T，设 u_1, u_2 是 u 的两个不同子结点，v 在去掉 u_1, u_2 的子树得到的树 T' 内，且 $u+v=u_1+u_2$。将 u_1, u_2 连接到 u 的边断开并接到 v 上，得到一棵新树 T''，如图所示：



因为 $|u_1-u|=|u+v-u_2-u|=|v-v_2|$，$|u_2-u|=|u+v-u_1-u|=|v-v_1|$，所以得到的新树能够按照原来的方式编号。

同理可以得到，设 u_1 是 u 的一个不同子结点，v 在去掉 u_1 的子树得到的树 T' 内，且 $u+v=2u_1$。将 u_1 连接到 u 的边断开并接到 v 上，得到一棵新树 T''，这棵新树能够按照原来的方式编号。记这两种操作为 $u\to v$ 移动。

选取一个点作为给定的树的根，使得根结点深度为 0 时每个点的深度不超过 2。对于根结点度数为奇数的情况，先将一个 n 个点的菊花的根结点编号为 n，其余结点顺次编号，容易说明编号合法。顺次执行操作 $n\to 1$，$1\to n-1$，$n-1\to 2$，$2\to n-2$，……，每次移动编号在关于 $(u+v)/2$ 对称的一段或两段区间中的点，留下编号不对称或离对称中心最远的点。这样的操作最终会在前几个移动到的位置上留下奇数个点，后面移动到的位置留下偶数个点（中间交替的位置就是保留 $(u+v)/2$ 不移动的位置），最后剩下一些没有移动到的位置。将给定的树的根的子结点按照度数分类，按照需要的树的样子执行以上操作即可。

对于根结点度数为偶数的非平凡情况，可以先忽略一棵子结点子树得到一种编号，然后将这些编号加一。接下来将刚刚忽略的子树的根编号为 1，再将其子结点顺次编号。容易发现编号依旧合法。

于是可以在 $O(n)$ 的时空复杂度内对任意一棵给定的树编号。