题目来源：「AGC034E」Complete Compress

算法一

输出 `-1`。

期望得分 12 分。

算法二 $n \le 2 \times 10^3$

枚举每个点作为最终的结点，以它为根做树形dp。

显然只会移动没有祖先关系的两个棋子。

设 siz_u 表示 u 的子树中有棋子的点的个数，dp_u 表示子树中能移动的步数的最大值，dis_u 表示子树中所有棋子到 u 的距离和。

如果 $2dp_u = dis_u$，则令 $ans = \min(ans, dp_u)$；没有满足条件的 u 则无解。

考虑如何转移，问题近似于给定一个序列 $a_1, a_2, \cdots, a_k$，每次选择两个正整数同时减一，求最多能操作多少步，也就是使最后剩下的那个正整数最小。

根据贪心，每次肯定选择最大的两个相消，设 mx 表示最大值，sum 表示所有数的和。

若 $mx > sum - mx$，说明最大值抵消完其它值后还有剩余，最后剩下的最小值一定是 $mx - (sum - mx)$。

若 $mx \le sum - mx$，操作后仍然保持这个状态，最后剩下的最小值一定是 0 或 1，

设 u 的儿子为 $v_1, v_2, \cdots v_k$，$a_{v_i} = dis_{v_i} + siz_{v_i}$。

那么转移就类似于序列上的问题了，只是当 $mx > sum - mx$ 时，剩下的 $mx - (sum - mx)$ 是可以在儿子的子树中继续移动的，此时就还需要考虑 dp_v 的贡献。

对每个点做一次树形dp，时间复杂度 $\mathcal{O}(n^2)$。

期望得分 60 分。

算法三

在算法二的基础上换根即可。

期望得分 100 分。