

字符修改 (string)

签到题，容易发现每一位是独立的，分别计算贡献。

对于一位，显然是把所有其他的字符，都变成一个字符，修改个数为 n 减去目标字符出现的次数，所以找一个出现次数最多的字符即可。

最大公约 (gcd)

首先有一个暴力做法，枚举连通块的 gcd ，然后找到所有 gcd 倍数的点，看看哪个连通块最大（可以使用并查集，或按照深度排序计算也可）。

然后我们发现，不需要枚举所有可能的 gcd ，只需要枚举质数即可算出所有答案，因为 $\text{gcd} > 1$ 就是至少有一个质数整除所有数，枚举这个质数即可。

由于每次枚举到一个质数 p ，设找到的是 p 的倍数的数有 t 个，那么时间复杂度是 $O(t \log t)$ 或 $O(t\alpha(t))$ 的，由于总共 n 个数的质因子个数是 $O(n \log V)$ 的，所以时间复杂度是 $O(n \log V \log n)$ 或 $O(n \log V \alpha(n))$ 的，均可通过。

图上加边 (graph)

首先，需要判断图中是否存在奇环和偶环的问题，判断是否出现了奇环，只需要黑白染色即可。

但是偶环是否存在不好判断，不过有一个性质：如果两个环有公共边，那么图中一定存在一个偶环。

证明考虑两环相交相当于两个点 a, b 之间有三条不交的路径，那么由于抽屉原理，一定有两条路径的长度奇偶性相同，故这两条路径连起来就能形成一个偶环。

所以，我们可以先对于每个连通块跑出一个 dfs 树出来，再考虑所有的返祖边，判断是否有奇环和直接的偶环，并做一遍树上前缀和求出每条边被多少条返祖边覆盖到，如果一条边被至少两条返祖边覆盖到，说明有两个环相交了，即出现了偶环。

接着，我们会发现，还需要计算是否存在长度为 $3/2$ 的链，这个可以先在 dfs 上求个直径，如果直径不超过 2 ，那么就是一个菊花的情况，再考虑上面如果有返祖边并且点数至少为 4 ，那么就有长度为 3 的链。

那么，最终答案为：

- $n \leq 4$: -1 ;
- 否则如果 $m \leq 2$: $5 - m$;
- 否则如果有偶环：
 - 有奇环: 0 ;
 - 无奇环: 1 ;

- 否则：
 - 有奇环：
 - 有长度为 3 的链：1；
 - 否则：2；
 - 无奇环：
 - 有长度为 3 的链：2；
 - 否则如果有度数至少为 3 的点：2；
 - 否则如果有长度为 2 的链：3；
 - 否则： $5 - \min(m, 2)$ 。

整个题目思维难度、算法难度、代码难度相当，略难于 NOIP T3，比较考验思维完整性。

疾速抓捕 (catch)

分两类情况讨论，一类是小偷速度大于等于警察，那么小偷一定会先向警察的方向走，直到快要被抓住了然后立刻回头走到距离最远的一个点去等警察来逮捕他；当小偷速度小于警察时，就会先向着警察走到尽可能远的一个点，然后快速向最远的点跑过去，让警察在到达最远点之前追不上他，或者可能是再多走一个点，让警察在一半追上他，两者都求一下取个较大值。

上述两种，都可以在路径上二分找到答案，树上路径的二分，需要先看看在 x 到 $LCA(x, y)$ 一段还是在 $LCA(x, y)$ 到 y 一段，然后再到对应的一边二分，可以写树上倍增，或者树剖加二分，std 使用了后者，常数较小。

另外，还需要预先做一遍换根 dp，求出 f_u 表示 u 往下走最远点距离，和 g_u 表示 f_{a_u} 不能往 u 里边走的最远点距离。代码细节不多，但是需要思考清楚再开始写，对代码能力提升是巨大的。