T1 评估（assess）

题目中所求解的 $|a_i - a_j|^2 = (a_i - a_j)^2$，这样我们可以不考虑绝对值。

对于 40% 的数据，由于此时 n 较小，我们可以根据题意进行模拟，枚举 i, j 后直接求得答案，时间复杂度 $O(n^2)$。

对于 100% 的数据，我们给出两种解法。

解法一：$(a_i - a_j)^2 = a_i^2 + a_j^2 - 2 \times a_i \times a_j$。于是我们将所求的答案拆分为 $a_i^2 + a_j^2$ 与 $-2 \times a_i \times a_j$ 两部分。前一部分 i 与 j 独立，我们只需要对每一个 i 求出它的贡献，即

$$\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} a_i^2 + a_j^2 = \sum_{i=1}^{n} a_i^2 \times (n-1)$$

对于后一部分

$$\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} -2 \times a_i \times a_j = \sum_{i=1}^{n-1} -2 \times a_i \times \sum_{j=i+1}^{n} a_j$$

j 求和的部分可以用前缀和求解，总时间复杂度 $O(n)$。

解法二：注意到 a_i 的值域较小，我们用桶记录每一个值出现的次数，接着考虑不同的桶之间相互的贡献。不妨设 i 出现的次数为 c_i，最后桶 i, j 之间的贡献为 $(i-j)^2 \times c_i \times c_j$。时间复杂度 $O(m^2)$（m 表示 a_i 的值域大小）。

T2 拆分数字（split）

容易发现，对于所有 n 都一定能表示（指用 $m_1, m_2, \ldots$），不过不一定是 k 个数。那我们可以求出最少和最多用多少个数表示，再看看能否累加或减少到 k 个。

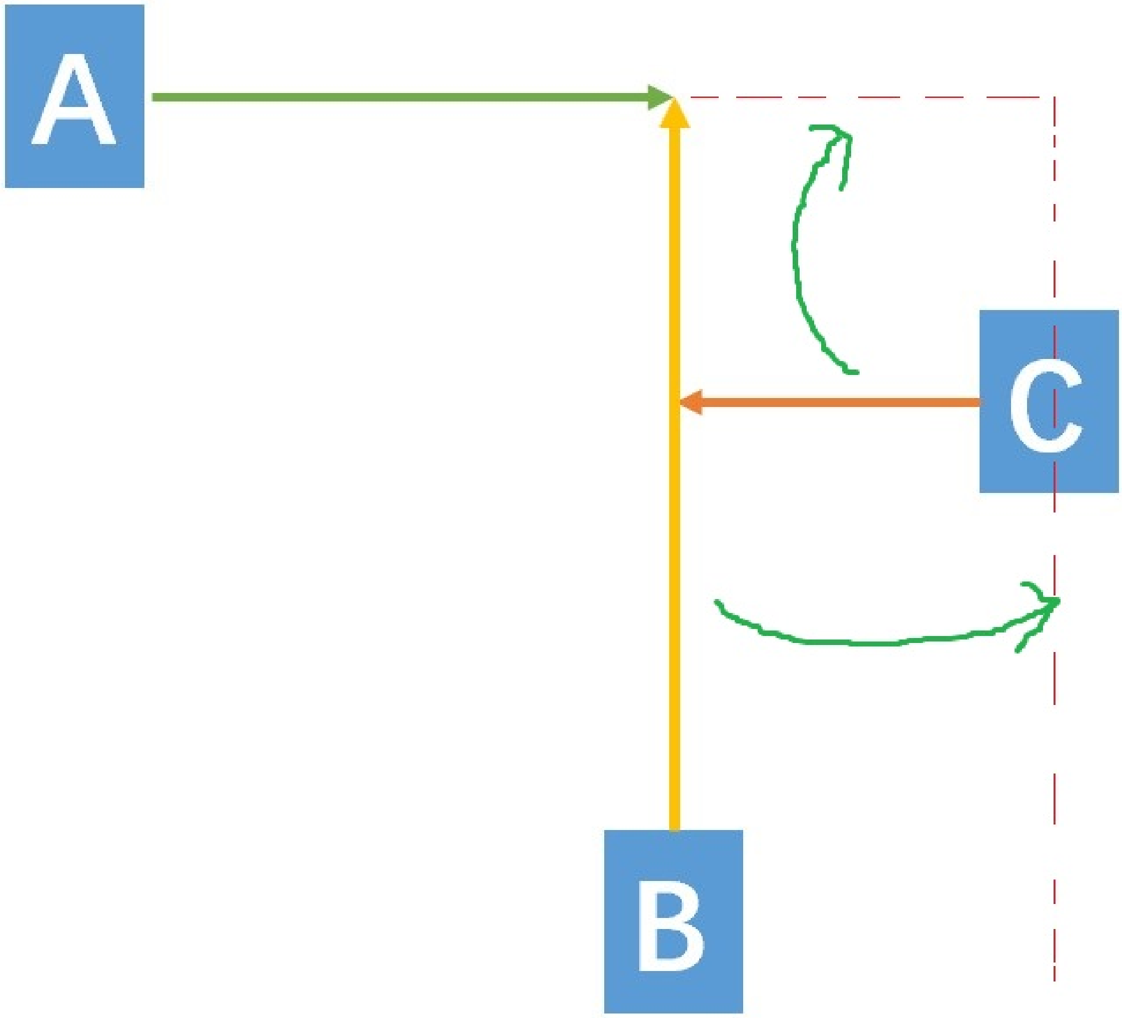
首先最多一定是 $n = 3^0 \times n$，最少则是 n 在三进制下的非零数的个数，不妨设其为 m。如果 $m > k$，一定无解，$m = k$ 有解。$m < k$ 则需要尝试拆分调整。

观察到我们可以进行这样的拆分：$3^i = 3^{i-1} + 3^{i-1} + 3^{i-1}$，即进行这样的一次拆分可以使原本的 m 增加 2。而 $n \geq k$，所有 m 一直拆分下去，一定有一个时刻 $\geq k$。只需要判断 m, k 的奇偶性是否相同即可。

T3 露营（camp）

这是一道贪心题。假如这是一个数轴而非平面上的问题，最少的清除个数即为最远的两点中间的所有小方格。

但若在平面上，我们可以分为 x, y 轴，合理猜想，答案为 x 方向上的答案加上 y 方向上的答案。



即类似上图的关系，我们可以通过适当的平移，将红虚线部分移至橙线与黄线部分。

T4 寻宝（treasure）

首先考虑没有法术的情况，那么很显然，直接从左上角起点开始往右和下找，每次都取当前次字典序更小的那个字母，如果有多个相同的，就全部考虑进去，在下一步中将所有的这些点当作起点一起考虑。其实就是类似使用 `BFS` 处理的分层最短路。

当我们使用里法术之后呢？很显然，为了使最终的答案字典序最小，肯定将从起点开始，将经过的所有非字母 a 的字母全部变为字母 a，这样得到的字典序最小。那么，我们将使用 k 次法术最远能够到的地方全部计算出来，再用 `BFS` 的方法后半部分内容，也能找到对应的答案，具体如下。

如何统计使用 k 次法术能到达的最远地方？我们可以使用 `dp`。定义 $f_{i,j}$ 表示从起点开始，到坐标 (i,j) 的路径上，最少非 a 字母数量。转移为 $f_{i,j}=\min(f_{i-1,j},f_{i,j-1})+[(i,j)\neq' a']$。

那么，只要 $f_{i,j}\le k$，我们就能使从起点开始到 (i,j) 的路径上最后所有的小写字母均为 a。而我们要字典序最小，就需要找出符合要求的答案里边的值 $i+j$ 最大的那一些（即路径最长，前缀 a 最多）。再从这些满足最大值的点开始进行 `BFS` 分层处理，就能找出最终字典序最小的序列。