divide

维护每个数的出现次数，按不断执行如下步骤：

- 取出出现次数最多的数，把它们放到一个集合里面。
- 然后不断取出出现次数最少的数，和上一步取出的数放在同一个集合里，直到放不下。

almostconvex

```
.0.
3#1
.2.
```

容易发现，对于一个凹包的中心格，其边缘格可以看成：在 $0,2$ 之间选一个，在 $1,3$ 之间选一个。

这样就是一个二分图的完美匹配计数问题，对于每个候选的边缘格看成一个节点，对于每个中心格，在 $0,2$ 对应的节点之间连一条边，在 $1,3$ 对应的节点之间连一条边（如果只有一个可用就连一个自环，如果没有可用的答案就是 0）。

对每个联通块数出完美匹配的方案数后，相乘，就是答案。

一个联通块的完美匹配，可以看成对每条边定一个向，满足所有节点的入度 ≤ 1。

- 一棵大小为 x 的树，你可以有 x 种定根方法，定完根之后连一个外向树，故有 x 种方案。
- 一棵基环树，如果环是自环，只有 1 种方案，否则环可以顺时针和逆时针，有 2 种方案。
- 其余情况，边数大于点数，0 种方案。

复杂度线性。

erase

考虑删完一个没有 `?` 的串 S 的条件，容易写出如下必要条件：

- $|S| \bmod (a+b) = 0$
- S 中有 $\frac{|S|}{a+b}a$ 个 `0` 和 $\frac{|S|}{a+b}b$ 个 `1`

事实上，它也是充分的，我们只需证明满足上述条件的串一定能执行至少一次操作。

将这个串分成长度为 $(a+b)$ 的 $\frac{|S|}{a+b}$ 块，如果有一块恰好有 a 个 `0`，就能执行一次操作，否则至少有相邻两段，一段 `0` 的个数 $< a$，另一段 $> a$，只需要把 $< a$ 的那一段往另一段挪，每挪一步，`0` 的个数的改变量的绝对值不会超过 1，故一定存在一个时刻 `0` 的个数 $= a$。

有 `?` 也是一样的，充要条件是：

- $|S| \bmod (a+b) = 0$
- S 中有不超过 $\frac{|S|}{a+b}a$ 个 `0` 和不超过 $\frac{|S|}{a+b}b$ 个 `1`

令 $dp(i)$ 表示前 i 个字符最多能删几次，转移：

- $dp(i) \to dp(i+1)$
- $dp(i) + k \to dp(i + k \cdot (a+b))$，这个转移需要满足 $(i, i + k \cdot (a+b)]$ 中有不超过 ka 个 `0` 和不超过 kb 个 `1`。

套一个 cdq 分治，第二种转移只会在 $\bmod k$ 相同的两个位置之间转移，故转移能写成二维偏序，复杂度 $O(n \log n)$。

cover

考虑模 D 意义下的限制，选择的矩形区域可以看做两个模 D 意义下的区间 $[L_x, R_x], [L_y, R_y]$，对于 A 类图形 (x, y) 可以看做 $x \in [L_x, R_x]$ 且 $y \in [L_y, R_y]$，对于 B 类图形 (x, y) 可以看做 $x \in [L_x, R_x]$ 或 $y \in [L_y, R_y]$。

枚举 $[L_x, R_x]$，将所有 A 类图形和目前不满足条件的 B 类图形的 y，放到一个长度为 D 的环上，需要找到一个最短的环上区间覆盖所有点，即求所有相邻两点之间的距离最大值。

枚举 L_x，如果能求出每个 y，在 R_x 扩大到什么时候，环上区间不需要覆盖 y（即没有 A 类图形有对应的 y，且所有拥有对应的 y 的 B 类图形已经被满足）。R_x 扩大的时候可能会有一些删除操作，用环状链表维护就行。

令 $T(L, y)$ 表示若 $L_x = l$，R_x 在什么时候会把 y 删除，观察到 $T(L, y)$ 在 L 增大的时候有单调性，用双指针预处理就行。

复杂度 $O(D^2)$。