算面积

矩阵和可以用二维前缀和计算，问题转化为给定下标 x, y 求二维前缀和。

由于每行的数是循环的，则每行的前缀和为循环节总和的若干倍加上循环节的某个前缀和。

同时，对于循环节长度相同的行，循环次数和余数都是相同的，因此，对于循环节长度相同的行，可以一起考虑。

将每种循环节长度分开，求其二维前缀和。查询时对每种循环节长度分别求二维前缀和，将它们加在一起就是答案。

时间复杂度 $O(l(r+q))$，其中 l 为循环节长度。

猜数

注意到 x 的位数与 z 相同或少一位，按两种反转方式分别搜索。

对称的位会相加，它们之间相互独立，方案数可以直接相乘。DFS 目的主要是判断可行性。

代码不太好写，要处理好进位问题。

排序

对于每个数 x，计算它会让几个数移到后面。

发现 $\geq x$ 的数不受影响；$< x$ 的数都会被移到最后。所以，值为 x 的若干个位置会贡献它们之间 $< x$ 数的个数，只有最后一个位置之后不受影响。用树状数组维护。

从大到小枚举 x。这个过程会移动所有 $< x$ 的数，是有后效性的。但在“移到最后”这个过程中，这些数在环上的相对位置不变。所以只需要维护环的起点。而由于上一轮已经全部移到最后，该轮环的起点就是上一轮的终点。

水池

注意到同一水平线上连续的水一定会一起流出，我们考虑对水池的区域进行划分。

对于每一条横向的边界，我们考虑将其延长，直至与纵向的边界相交。设横向边界的条数为 m，则水池划分为 m 个区域。

现在我们考虑一个区域中的水如果全部流出会造成什么影响，会使上面一部分的水也流出。

而一个区域上面所连接的区域底边，则是这个区域底边往左往右第一个大于它的高度的较小的一侧。即所有区域按横向边界的高度形成一棵笛卡尔树。

我们将笛卡尔树建出，则题意转化为选择 k 个节点，答案即为这些节点到根的链的并集的权值和，这是一个经典问题。

这里不加证明地给出两种做法：

- 可以发现贪心是正确的，因此我们用数据结构维护每个叶子当前到根的权值和，每次贪心选取权值和最大的一个叶子并修改即可。
- 基于贪心的思路，我们可以使用类似于长链剖分的算法，直接求出最终取得每条链的位置，然后排序取前 k 大即可。

猜数题解2

对于一个数 x 我们考虑将他按位分解成形如 $x_1, x_2, x_3 \dots$ 的数列

同理，我们也将 z 分解为 $z_1, z_2, z_3 \dots$

那么 $x + mirrored(x)$ 在不进位下为 $x_1 + x_{len}, x_2 +_{len-1}, x_3 + x_{len-2} \dots$

容易发现 $x_1 + x_{len}, x_2 + x_{len-1}, x_3 + x_{len-2} \dots$ 进位后就是 $z_1, z_2, z_3 \dots$

由此，我们可以得到两个信息：

1 答案 x 与 z 的位数 len_x, len_z 的差的绝对值最大为 1

2 $x + mirrored(x)$ 在不进位之下的状态最多只有 $2^{(len-1)>>1}$ 种

所以我们考虑枚举每一位的进位情况，因为是回文的，所以我们只用枚举一半即可

考虑判断我们枚举的不进位的 $x + mirrored(x)$ 是否可能为答案

其实只需要进位后看相不相同即可

统计答案时，我们计算对于 $x + mirrored(x)$ 在不进位之下，每一位能取得上界和下界

因为互相独立，所以直接乘起来即可

注意， x 的开头不能为 0