

异或运算(xor)

【问题描述】

小 Q 是一位勤奋的小学生，他的理想是进入自己心仪的大学学习计算机专业。为了实现这一目标，他从小就开始认真学习信息学竞赛的基础知识。

今天，小 Q 学习了异或运算。为了检验自己是否熟练掌握了异或运算，小 Q 决定给自己出一道题。他随机生成了数列 $X = \{x_1, x_2, \dots, x_n\}$ 和数列 $Y = \{y_1, y_2, \dots, y_m\}$ 。对于数列 X 和数列 Y 中的每一对数 (x_i, y_j) ，小 Q 都会计算它们按位异或的结果，并将其填写到矩阵 A 中的第 i 行第 j 列，即：

$$A_{ij} = x_i \text{ xor } y_j$$

得到这个 n 行 m 列的矩阵 A 后，小 Q 需要检验自己的计算结果是否正确。检验所有异或运算的结果过于繁琐，所以小 Q 想了一个简便的检验方法。他会在矩阵 A 中选择一个矩形区域，并找到这个矩形区域中第 k 大的数，若这个数与标准答案一致，即通过了一次检验。小 Q 一共要进行 p 次检验，若 p 次检验全部通过，他就认为计算的矩阵 A 是基本正确的。

但是，小 Q 现在并没有标准答案。所以，他向你求助，希望你编写一个程序，帮他计算出每次检验的标准答案。

【输入格式】

第一行包含两个正整数 n, m ，分别表示两个数列的长度；

第二行包含 n 个非负整数 x_i ，含义见题目描述；

第三行包含 m 个非负整数 y_j ，含义见题目描述；

第四行包含一个正整数 p ，表示检验的次数；

随后 p 行，每行均包含五个正整数，用来描述一次检验；

其中，第 i 行包含的前四个正整数为 U_i, D_i, L_i, R_i ，分别表示第 i 次检验区域的上、下、左、右边界（检测的矩形区域包含边界）；第五个正整数为 K_i ，表示此次检验的是该矩形区域中的第 K_i 大数。

【输出格式】

共 p 行，每行仅包含一个非负整数，表示此次检验的标准答案。

【样例输入】

```
3 3
1 2 4
7 6 5
3
1 2 1 2 2
1 2 1 3 4
2 3 2 3 4
```

【样例输出】

6
5
1

【样例数据说明】

通过计算可得到矩阵如下：

6	7	4
5	4	7
3	2	1

第一次寻找 (6, 7, 5, 4) 中的第 2 大值, 答案为 6;

第二次寻找 (6, 7, 4, 5, 4, 7) 中的第 4 大值, 答案为 5;

第三次寻找 (4, 7, 2, 1) 中的第 4 大值, 答案为 1。

【数据规模与约定】

对于全部测试数据, 均满足:

$$1 \leq U_i \leq D_i \leq n \leq 1000;$$

$$1 \leq L_i \leq R_i \leq m \leq 300,000;$$

$$0 \leq x_i, y_i < 2^{31};$$

$$1 \leq K_i \leq (D_i - U_i + 1) \times (R_i - L_i + 1);$$

$$1 \leq p \leq 500。$$

其中, 部分测试数据有如下特征 (互不包含):

对于 5% 的数据, 满足 $1 \leq m \leq 1,000$, $1 \leq p \leq 10$, $K_i = 1$;

对于 15% 的数据, 满足 $1 \leq m \leq 3,000$, $1 \leq p \leq 200$;

对于 20% 的数据, 满足 $p = 1$;

对于 30% 的数据, 满足 $K_i = 1$;

对于其余 30% 的数据, 没有其他特征。