

题目名称	晓月的数学题	异或路径	子串
题目类型	传统型	传统型	传统型
目录	math	xor	substr
可执行文件名	math	xor	substr
输入文件名	math.in	xor.in	substr.in
输出文件名	math.out	xor.out	substr.out
每个测试点时限	1.0秒	1.0秒	1.0秒
内存限制	256MiB	256MiB	256MiB
测试点数目	20	20	20
测试点是否等分	是	是	是
提交源代码文件名	math.cpp	xor.cpp	substr.cpp

晓悦的数学题

给定一个自然数 x ，求解 3^x 的个位是多少。具体理解可看样例。

输入格式

第一行输入一个自然数 x 。

输出格式

输出一个数字代表答案。

样例 #1

样例输入 #1

3

样例输出 #1

7

样例 #2

样例输入 #2

2

样例输出 #2

9

样例 #3

样例输入 #3

0

样例输出 #3

1

提示

规范

对于 50% 的测试数据 $0 \leq x \leq 15$

对于 100% 的测试数据 $0 \leq x \leq 2^{31}$

Sample Explanation 1

对于样例1, $3^3 = 7$, 则个位为 7。

异或路径

题目描述

给定一个 N 行 N 列的网格，用 (i, j) 表示从顶部开始的第 i 行和从左侧开始的第 j 列的单元格（ $1 \leq i, j \leq N$ ）。每个单元格 (i, j) 中写有一个非负整数 $a_{i,j}$ 。

当你位于单元格 (i, j) 时，你可以移动到单元格 $(i + 1, j)$ 或 $(i, j + 1)$ 中的任何一个。但是，你不能移动到网格外面。

求从单元格 $(1, 1)$ 开始，通过重复移动到达单元格 (N, N) 的方法中，经过的单元格（包括单元格 $(1, 1)$ 和 (N, N) ）上写的整数的异或和为 0 的路径的总数是多少？

输入格式

第一行包含一个整数 N ，表示网格的行数和列数。

接下来的 N 行，每行包含 N 个非负整数，表示网格中每个单元格的值。每个整数 $a_{i,j}$ 满足 $1 \leq a_{i,j} \leq 8$ 。

输出格式

输出一个整数代表答案

样例 #1

样例输入 #1

```
3
1 5 2
7 0 5
4 2 3
```

样例输出 #1

```
2
```

样例 #2

样例输入 #2

```
2
1 2
2 1
```

样例输出 #2

```
0
```

样例 #3

样例输入 #3

```
10
1 0 1 0 0 1 0 0 0 1
0 0 0 1 0 1 0 1 1 0
1 0 0 0 1 0 1 0 0 0
0 1 0 0 0 1 1 0 0 1
0 0 1 1 0 1 1 0 1 0
1 0 0 0 1 0 0 1 1 0
1 1 1 0 0 0 1 1 0 0
0 1 1 0 0 1 1 0 1 0
1 0 1 1 0 0 0 0 0 0
1 0 1 1 0 0 1 1 1 0
```

样例输出 #3

```
24307
```

提示

数据范围

对于 20% 的测试数据 $2 \leq N \leq 10$.
对于 50% 的测试数据 $2 \leq N \leq 15$.
对于 100% 的测试数据 $2 \leq N \leq 20$.

Sample Explanation 1

有以下两种方案实现

$$\begin{aligned} &(1, 1) \rightarrow (1, 2) \rightarrow (1, 3) \rightarrow (2, 3) \rightarrow (3, 3) \\ &(1, 1) \rightarrow (2, 1) \rightarrow (2, 2) \rightarrow (2, 3) \rightarrow (3, 3) \end{aligned}$$

子串

题目描述

你有一个数列 a ，其中 $1 \sim n$ 各出现了一次。

当你任意选一对 $1 \leq l \leq r \leq n$ ，并将 a_l, a_{l+1}, \dots, a_r 排成一行，你就得到了 a 的一个子串，记为 $a_{l \sim r}$ ，称 l 为左端点， r 为右端点。

你需要把 a 所有子串按字典序从小到大排序。但是为了避免输出量过大，我会给出 q 个问题，每次给出一个 k ，求字典序第 k 小的子串左右端点。

如果你不知道什么是字典序，看这里：

对于两个数列 p, q ，称 p 的字典序小于 q （记为 $p < q$ ），当且仅当存在自然数 k 使 p, q 的前 k 个数相同且 $p_{k+1} < q_{k+1}$ 。

特别地，若 p 是 q 的前缀且 $p \neq q$ ，也认为 p 的字典序小于 q 。

例如：

- $[1, 2] < [3, 2]$ (当 $k = 0$)
- $[3, 1, 100] < [3, 2, 1]$ (当 $k = 1$)
- $[3, 4] < [3, 4, 6]$ (p 是 q 前缀)

输入格式

输入的第一行有两个正整数 n, q 表示序列长度和询问个数。

第二行有 n 个正整数 a_1, a_2, \dots, a_n ，表示这个数列。

之后有 q 行，每行有一个正整数 k ，表示要求的子串的排名。

输出格式

对于每个问题，输出一行两个整数 l, r ，表示字典序第 k 小的子串是 $a_{l \sim r}$ 。

样例 #1

样例输入 #1

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

样例输出 #1

```
2 2
2 3
3 3
1 1
1 2
1 3
```

样例 #2

样例输入 #2

```
50 25
42 22 27 8 44 11 14 31 37 10 48 15 12 40 13 4 25 9 19 5 2 18 6 1 32 3 38 33 43 34
46 47 23 35 21 20 45 39 50 7 36 17 24 29 16 30 49 26 28 41
1178
991
755
1094
689
132
671
635
421
659
448
334
327
213
1206
453
1160
583
388
781
150
692
23
1162
62
```

样例输出 #2

```
37 48
27 44
3 28
1 46
43 47
20 34
33 37
2 19
15 44
```

2 43
7 27
6 31
6 24
4 29
32 37
7 32
5 44
19 47
13 47
44 45
23 24
43 50
24 46
5 46
26 30

提示

【样例解释 #1】

数列 3, 1, 2 共有 6 个子串, 从小到大排序的结果为: [1], [1, 2], [2], [3], [3, 1], [3, 1, 2]。

【数据范围】

测试点编号	$n \leq$	$q \leq$	特殊性质
1 ~ 3	200	200	
4 ~ 7	1000	3×10^5	
8 ~ 9	3000	3×10^5	
10 ~ 13	3×10^5	10	
14 ~ 15	3×10^5	3×10^5	$a_i = i$
16 ~ 20	3×10^5	3×10^5	

对于全体数据, 保证 $1 \leq n, q \leq 3 \times 10^5$, $1 \leq k \leq \frac{n(n+1)}{2}$, a_i 中 $1 \sim n$ 各有一个, 输入皆为整数。