

P84

竞赛时间：????年??月??日?:?-?:??

题目名称	a	b	c
名称	a	b	c
输入	stdin	stdin	stdin
输出	stdout	stdout	stdout
每个测试点时限	1 秒	2 秒	1 秒
内存限制	512MB	512MB	512MB
测试点数目	20	20	20
每个测试点分值	5	5	5
是否有部分分	无	无	无
题目类型	传统	传统	传统

注意事项（请务必仔细阅读）：



a

【问题描述】

组合数 C_n^m 表示的是从 n 个物品中选出 m 个物品的方案数。举个例子，从 $(1,2,3)$ 三个物品中选择两个物品可以有 $(1,2), (1,3), (2,3)$ 这三种选择方法。根据组合数的定义，我们可以给出计算组合数 C_n^m 的一般公式：

$$C_n^m = \frac{n!}{m!(n-m)!}$$

其中 $n! = 1 \times 2 \times \cdots \times n$ 。（额外的，当 $n = 0$ 时， $n! = 1$ ）

小葱想知道如果给定 n, m 和 k ，对于所有的 $0 \leq i \leq n, 0 \leq j \leq \min(i, m)$ 有多少对 (i, j) 满足 C_i^j 是 k 的倍数。

【输入格式】

第一行有两个整数 t, k ，其中 t 代表该测试点总共有多少组测试数据， k 的意义见【问题描述】。

接下来 t 行每行两个整数 n, m ，其中 n, m 的意义见【问题描述】。

【输出格式】

t 行，每行一个整数代表所有的 $0 \leq i \leq n, 0 \leq j \leq \min(i, m)$ 有多少对 (i, j) 满足 C_i^j 是 k 的倍数。

【样例输入 1】

```
1 2
3 3
```

【样例输出 1】

```
1
```

【样例解释 1】

在所有可能的情况中，只有 C_2^1 是2的倍数。

【样例输入 2】

```
2 5
4 5
6 7
```

【样例输出 2】

0
7

【数据规模与约定】

对于20%的测试点， $1 \leq n, m \leq 100$;

对于另外15%的测试点， $n \leq m$;

对于另外15%的测试点， $k = 2$;

对于另外15%的测试点， $m \leq 10$;

对于100%的测试点， $1 \leq n, m \leq 10^{18}, 1 \leq t, k \leq 100$ ，且 k 是一个质数。

b

【问题描述】

小 H 是一位勤奋的中学生，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 H 学习了平方运算。为了检验自己是否熟练掌握了平方运算，小 H 决定给自己出一道题。小 H 有一个长度为 N 的序列 $\{X_1, X_2, \dots, X_N\}$ 。小 H 会时不时地取出序列中的一段连续区间 $[l, r]$ ，并将其中的每一个数改为原数值的平方对 p 取模的结果，即

$$\forall i \in [l, r], X_i \leftarrow (X_i \times X_i) \bmod p$$

其中， p 为某个给定的数。为了检验自己的运算是否正确，小 H 还会时不时地想要知道序列中某一段连续区间 $[l, r]$ 内所有数的和是多少。

但是，小 H 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，帮他计算出每次想要知道的区间内的数的总和。

【输入格式】

第一行有三个整数 N, M, p ，分别代表序列的长度、平方操作与询问操作的总次数以及在平方操作中所要模的数。

接下来一行 N 个数代表一开始的序列 $\{X_1, X_2, \dots, X_N\}$ 。

接下来 M 行，每行三个整数 op, l, r 。其中 op 代表本次操作的类型。若 $op = 0$ ，代表这是一次平方操作，平方的区间为 $[l, r]$ ；如果 $op = 1$ ，代表这是一次询问操作，询问的区间为 $[l, r]$ 。

【输出格式】

对于每次的询问操作，输出一行代表这段区间内数的总和。注意：答案没有对任何数取模。

【样例输入 1】

```
3 3 11
1 2 3
1 1 3
0 1 3
1 1 3
```

【样例输出 1】

```
6
14
```

【样例输入 2】

```

3 3 3
0 1 2
1 1 3
0 1 3
1 1 3

```

【样例输出 2】

```

3
2

```

【数据规模与约定】

对于 100% 的数据， $\forall i, X_i \in [0, p), l, r \in [1, N]$ 。

N, M, p 的范围详见下表：

编号	N	M	p
1	$\leq 1,000$	$\leq 1,000$	$= 233$
2			$= 2332$
3	$\leq 100,000$	$\leq 100,000$	$= 5$
4			$= 8192$
5	$\leq 100,000$	$\leq 100,000$	$= 23$
6			$= 45$
7			$= 37$
8	$\leq 55,000$	$\leq 55,000$	$= 4185$
9			$= 5850$
10			$= 2975$
11			$= 2542$
12			$= 2015$
13	$\leq 60,000$	$\leq 60,000$	$= 2003$
14	$\leq 65,000$	$\leq 65,000$	$= 2010$
15	$\leq 70,000$	$\leq 70,000$	$= 4593$
16	$\leq 75,000$	$\leq 75,000$	$= 4562$
17	$\leq 80,000$	$\leq 80,000$	$= 1034$
18	$\leq 85,000$	$\leq 85,000$	$= 5831$
19	$\leq 90,000$	$\leq 90,000$	$= 9905$
20	$\leq 100,000$	$\leq 100,000$	$= 9977$

C

【问题描述】

小 J 是一位勤奋的大学生，在清华大学计算机系学习的他，每天都会遇到不少有挑战性的问题。

今天，小 J 的老师在课上讲了一个字符串的加密算法，对于一个长度为 N 的字符串，我们在字符串的末尾添加一个特殊的字符“.”。之后将字符串视作一个环，从位置 $1, 2, 3, \dots, N+1$ 为起点读出 $N+1$ 个字符，就能得到 $N+1$ 个字符串。比如对于字符串“ABC AAA”，我们可以得到这 $N+1$ 个串：

```

ABC AAA.
BC AAA. A
C AAA. AB
AAA. ABC
AA. ABCA
A. ABCAA
. ABCAAA

```

接着我们对得到的这 $N+1$ 个串按字典序从小到大进行排序（注意特殊字符“.”的字典序小于任何其他字符）结果如下：

```

. ABCAAA
A. ABCAA
AA. ABCA
AAA. ABC
ABC AAA.
BC AAA. A
C AAA. AB

```

最后，将排序后的 $N+1$ 个串的最后 N 个字符取出，按照顺序排成一个新的字符串，也就是上面这个表的最后一列，就是加密后的密文“AAAC. AB”。

聪明的小 J 很快就理解了加密算法，然而因为课堂的时间有限，老师没有来得及讲解密算法就下课了。好奇的小 J 很想知道如何对字符串进行解密，即通过加密后的密文求出加密前的字符串。你能帮他解决这个问题吗？

【输入格式】

第一行有两个整数 N, M ，分别表示加密前的字符串长度和字符集的大小，其中字符用整数 $1, 2, 3, \dots, M$ 编号，添加的特殊字符“.”用 0 编号。

第二行为 $N+1$ 个整数，表示加密后的字符串。

【输出格式】

输出仅一行，包含 N 个整数，用空格隔开，依次表示加密前字符串中每个字符的编号。

【样例输入】

6 3
1 1 1 3 0 1 2

【样例输出】

1 2 3 1 1 1

【对样例的说明】

将样例输入与输出中的 1, 2, 3 分别视为 A, B, C，则样例即为题目描述所述的字符串。

【数据规模与约定】

测试数据的规模如下表所示

编号	N	M	特别约定
1	= 10	≤ 3	无
2	= 15		
3	= 20		
4	= 25		
5-6	≤ 50	≤ 50	字符串中的字符互不相同
7-8	$\leq 1,000$	$\leq 1,000$	
9-12			无
13-20	$\leq 200,000$	$\leq 200,000$	