

2023NOIP 模拟赛

题目名称	小D的序列	小S排座位	小K的外挂	小Z的作业
题目类型	传统型	传统型	传统型	传统型
目录	sequence	seat	cheat	darkduck
源程序文件名	sequence.cpp	seat.cpp	cheat.cpp	darkduck.cpp
可执行文件名	sequence	seat	cheat	darkduck
输入文件名	sequence.in	seat.in	cheat.in	darkduck.in
输出文件名	sequence.out	seat.out	cheat.out	darkduck.out
每个测试点时限	1s	1s	1s	2s
内存限制	1024MB	1024MB	1024MB	1024MB
子任务数目	4	6	5	7

1. T1可能不是最简单的题。
2. 需要建立子文件夹。
3. 子任务的依赖与数据限制的包含关系相同（即默认依赖）。
4. 评测在 Linux 下进行。
5. 文件名（包括程序名和输入输出文件名）必须使用英文小写。
6. 结果比较方式为忽略行末空格、文末回车后的全文比较。
7. C/C++ 中函数 main() 的返回值类型必须是 int，值为 0。
8. 编译选项为-O2 -std=c++14。
9. 选手提交的程序源文件必须不大于 100KB。
10. 题面不一定包含全部样例，请查看下发附加文件以得到所有样例。
11. 如果对题目有疑问（如样例出锅），可以找出题人。
12. 题目过于AK，请喧哗后不要大声简单。

小D的序列

题目描述

小D今天学习了哈希，因此她对一个等比数列在模意义下的值产生了浓厚兴趣。她选定了三个数 n, a, p ($0 < a < p$)，并生成了一个共 n 项且下标从 0 开始的序列 $f_i = a^i \bmod p$ 用于研究。但可惜的是，她不小心把 f 排序了一下，并且她忘记了原先的 a 的值，请你告诉她原先的 a 等于几。

注：题目描述中并没有包含部分限制条件，请仔细阅读**限制与约定**。

输入格式

第一行输入两个数 n, p ，表示数组长度和模数。

第二行输入 n 个数，第 i 个数为 f'_{i-1} ，其中序列 f' 是序列 f 排序后的内容。

输出格式

一行一个整数 a ，表示原定的底数。

样例○

input

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

output

```
3
```

样例○解释

该样例不符合数据范围限制，仅用于自测。

样例一

input

```
10 998244353
1 2 4 8 16 32 64 128 256 512
```

output

```
2
```

样例二

input

```
20 135444259
1 3362685 13531079 15624696 18351432 18603929 23198186 26790643 41443981 42602041
47035008 49861672 55270056 59850459 71138775 86446610 111025622 123530178
124170373 126387466
```

output

```
42602041
```

限制与约定

对于 100% 的数据，保证 $2 \leq n \leq 2 \times 10^5$ ， $10^8 \leq p \leq 10^9$ ，保证 f' 的元素互不相同且不包含 0，保证 p 是质数，保证答案有唯一解。

子任务编号	分值	$n \leq$	特殊限制
1	5	2	$p = 998244353$
2	30	10^3	保证 2 是质数
3	50	10^5	保证 a 和 p 随机
4	15	2×10^5	保证 $1 + 1 = 2$

小S排座位

题目描述

作为学校违纪行为的代言人，小S被老师安排去排一个新班级的座位。新班级有 n 位同学，他们的成绩按顺序分别为 a_1, a_2, \dots, a_n ($a_i \leq a_{i+1}$)。教室的一桌会坐相邻的两个人，称为同桌。为促进互帮互助，一对同桌的成绩之差的绝对值必须**大于等于**给定的常数 K ，但这样会导致一些同学没有同桌，而没有同桌的人会去暴D小S。小S不想被D，所以请你求出他最多能排出几对同桌。

输入格式

第一行输入两个整数 n, K ，含义见题目描述。

第二行输入 n 个整数 a_i ，保证数组 a 不降。

输出格式

第一行一个整数 ans ，表示答案。

样例一

input

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

output

```
2
```

样例二

input

```
8 5
1 4 5 6 8 9 10 15
```

output

```
4
```

限制与约定

对于100%的数据，保证 $1 \leq n \leq 10^6, 0 \leq a_i, K \leq 10^9$

子任务编号	分值	$n \leq$	$a_i, K \leq$	特殊性质
1	5	2	10^9	无
2	15	100	10^9	无
3	15	10^6	100	无
4	15	10^6	10^9	保证 $a_i = i$
5	15	10^6	10^9	保证 $K > \frac{a_n}{2}$
6	35	10^6	10^9	无

小K的外挂

题目描述

小K是一个挂B，他很喜欢开挂，但是有一天，他的挂坏了，这让他很是头疼。

小K在下飞行棋，棋盘是一行共 n 个格子，编号依次为 $1 \sim n$ 。小K有外挂，所以他会通过外挂来前进，他共有 m 个外挂，第 i 个外挂能让他从第 l_i 格瞬移到第 r_i 格，并花费 1 的时间，并且小K很D，所以他不需要外挂就能往回走（即从第 i 格走到第 $i - 1$ 格，且往回走不需要时间）。

但是小K的 rp 不太好，现在发生了 q 次事件，每次事件中，小K的某一个挂会坏（这次事件结束后又会恢复），而你需要帮他求出此时他从 s 走到 t 要花费的最小时间。

输入格式

第一行两个整数 n, m ，表示格子数量和外挂数量。

接下来 m 行，每行两个整数 l_i, r_i ，表示一个外挂。

接下来一个整数 q ，表示询问数量。

接下来 q 行，每行三个整数 id, s, t ，表示第 id 个挂坏了，你需要求出从 s 走到 t 所需的最小时间。

输出格式

输出共 q 行，每行一个整数表示最小时间，如果无法到达，输出 -1 。

样例一

input

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

output

```
-1
1
2
```

样例二

input

```
10 6
1 3
2 2
2 4
3 8
5 7
6 10
5
1 1 2
3 1 4
4 2 8
6 2 8
5 2 9
```

output

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

限制与约定

对于 100% 的数据 , 满足 $1 \leq n, m, q \leq 2 \times 10^5$, $1 \leq l_i \leq r_i \leq n$, $1 \leq s, t \leq n$, $1 \leq id \leq m$ 。

子任务编号	分值	$n, m \leq$	$q \leq$	特殊性质
1	5	1000	10	无
2	15	1000	2×10^5	无
3	20	2×10^5	20	无
4	20	2×10^5	2×10^5	$r_i - l_i \leq 5$
5	40	2×10^5	2×10^5	无

小Z的作业

题目描述

教练给小Z布置了一个作业，教练给出了 n 个命题，让他证明 n 个命题全部为真。但是小Z非常的懒，一旦需要证明的命题数量大于 K ，小Z就会摆烂不完成作业。因此小Z向他的好朋友黑恶卷怪小L求助。小L帮小Z证明了 m 个等价关系，即对于 $1 \leq i \leq m$ ，小L证明了 u_i 和 v_i 是等价的。小L把他的成果发给了小Z，但是网络质量不佳，小Z只收到了一个区间的等价关系证明。

现在有 q 条时间线，你需要对于每条时间线，求出假如小Z收到了第 $l \sim r$ 个等价关系的证明，小Z是否能够完成作业。

输入格式

第一行四个整数 n, m, K, tp ，分别表示小Z需要证明的命题个数，小L证明出的等价关系数，小Z摆烂的临界值，是否强制在线。

接下来 m 行，第 $i + 1$ 行包含两个整数 u_i, v_i ，表示小L的第 i 个证明给出了 u_i 和 v_i 是等价的结论。

第 $m + 2$ 行一个整数 q ，表示时间线的个数。

接下来 q 行，每行两个整数 l_i, r_i ，表示小Z收到了第 $l_i \sim r_i$ 个证明，你需要求出小Z是否能够完成作业。如果小Z能够完成作业，输出 "Yes"（不包含引号），如果小Z会摆烂不完成作业，输出 "No"（不包含引号）。

如果 $tp = 1$ ，表示输入的 l_i 和 r_i 是经过加密的。用 $lans_{i-1}$ 表示 $1 \sim i - 1$ 个询问的答案按顺序从左到右拼接构成的二进制数（Yes视为 1，No视为 0），你需要令

$$l_i \leftarrow (((l_i + lans_{i-1}) \bmod 2^{32}) \bmod m) + 1,$$
$$r_i \leftarrow (((r_i + lans_{i-1}) \bmod 2^{32}) \bmod m) + 1,$$

然后如果 $l_i > r_i$ 则交换 l_i, r_i 。

以下参考代码片段实现了这一强制在线的过程：

```
unsigned lans=0;
for(int i=1;i<=q;++i){
    int l,r;
    std::cin>>l>>r;
    if(tp==1){
        l=(l+lans)%m+1;
        r=(r+lans)%m+1;
        if(l>r) std::swap(l,r);
    }
    //solve
    lans<<=1;
    if(Answer_Is_Yes) ++lans;
}
```

本题下发文件中的 `online.cpp` 会包含该代码段。

输出格式

每行一个字符串 Yes 或者 No，表示小Z是否能够完成作业。

样例一

input

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

output

```
Yes
Yes
Yes
No
Yes
```

样例二

input

```
10 10 6 1
3 9
2 3
7 10
5 3
2 10
4 8
7 6
3 2
4 3
10 10
10
6 7
9 10
3 8
6 7
3 7
1 3
7 10
3 5
```

```
4 9
5 6
```

output

```
No
Yes
Yes
Yes
Yes
No
Yes
No
Yes
No
```

样例解释二

加密前的原始数据为

```
7 8
1 10
5 10
1 10
1 5
7 9
1 8
5 7
2 7
1 2
```

限制与约定

对于 100% 的数据，满足 $1 \leq n, m \leq 2 \times 10^5$ ， $1 \leq K \leq n$ ， $tp \in \{0, 1\}$ ， $1 \leq q \leq 5 \times 10^5$ ， $1 \leq u_i, v_i \leq n$ ， $1 \leq l_i, r_i \leq m$ ，当 $tp = 0$ 时，有 $l_i \leq r_i$ 。

子任务编号	分值	$n, m \leq$	$q \leq$	$tp \in$	特殊性质
1	5	2×10^5	5×10^5	$\{0, 1\}$	$K = 1$ 且 $n > m + 1$
2	5	2×10^5	5×10^5	$\{0, 1\}$	$K = n$
3	10	2×10^3	5×10^5	$\{0\}$	无
4	10	2×10^5	10^3	$\{0, 1\}$	无
5	25	2×10^5	5×10^5	$\{0\}$	无
6	15	10^5	10^5	$\{0, 1\}$	无
7	30	2×10^5	5×10^5	$\{0, 1\}$	无

