

CSP2023模拟赛

题目名称	博弈	二进制	聚众斗殴	可怜
题目类型	传统型	传统型	传统型	传统型
可执行文件名	act	gen	fight	kel
输入文件名	act.in	gen.in	fightin	kel.in
输出文件名	act.out	gen.out	fight.out	kel.out
每个测试点时限	2.0 秒	1.0 秒	1.0 秒	1.0 秒
内存限	1024 MiB	1024 MiB	1024 MiB	1024MiB
子任务/测试点数目	6	3	5	20
是否等分	否	否	否	是

提交源文件程序名

对于C++语言	act.cpp	gen.cpp	fight.cpp	kel.cpp
---------	---------	---------	-----------	---------

编译选项

对于C++语言	-lm -O2 -std=c++17
---------	--------------------

注意事项(请仔细阅读)

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C/C++中函数main()的返回类型必须是int，程序正常结束时返回值必须是0。
3. 选手提交的程序代码文件请在**个人目录下以及子文件夹内各放一份**。
4. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
5. 选手提交的程序源文件必须不大于100KB。
6. 程序可使用的栈空间内存限制于题目的内存限制一直。
7. 使用std::deque等STL容器时，请注意其内存空间消耗。
8. 评测时采用的机器配置为 AMD Ryzen 7 5800H with Radeon Graphics，内存16GiB。上述时限以此配置为准。
9. 评测在Windows 10下进行，使用LemonLine进行评测。

博弈 (act)

【题目描述】

Alice 和 Bob 初始有两个数 a, b 。Alice 和 Bob 轮流进行以下的操作，Alice 为先手：

- 选择一个数，把它减去任意一个不超过它的正整数。

当一个人操作前，出现如下情况之一，则操作方输：

1. a, b 中出现负数；
2. $a = b = 0$ ；
3. Alice 和 Bob 事先选定了 $2n$ 个数 $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n$ ，存在一个 $i \in [1, n]$ 使得 $a = a_i \wedge b = b_i$ 。

现在，Alice 和 Bob 决定进行 q 局博弈。众所周知，Alice 和 Bob 是绝顶聪明的，所以，请你预测一下他们的博弈结果。

【输入格式】

第一行两个非负整数 n, q 。

第二行 n 个非负整数表示 a_1, a_2, \dots, a_n 。

第三行 n 个非负整数表示 b_1, b_2, \dots, b_n 。

接下来 q 行，每行两个数 a, b 表示一局博弈中初始的两个数。

【输出格式】

共 m 行，每行输出 Alice 或 Bob 表示胜者。

【输入输出样例1】

act.in	act.out
3 3	
2 3 3	Alice
1 1 4	Bob
0 5	Alice
3 1	
6 6	

【输入输出样例1说明】

对于第一局游戏，Alice 可以把 b 减去 5，使得 $a = b = 0$ ，因此 Bob 会输。

对于第二局游戏，开始时， $a = a_2, b = b_2$ ，故 Alice 一开始就会输。

对于第三局游戏，容易手玩得到 Alice 一定有必胜策略。

【数据规模与约定】

对于所有测试数据, 满足 $0 \leq n, q \leq 10^6, 0 \leq a, b, a_i, b_i \leq 10^9$, 保证二元组 (a_i, b_i) 互不相同且 a_i, b_i 不同时为 0;

子任务编号	特殊性质	分值
1	$q = 0$	1
2	$n = 0$	6
3	$n = 1$	12
4	$a, b, a_i, b_i \leq 3 \times 10^3$	19
5	$n, q \leq 2 \times 10^3$	19
6	无	43

二进制 (gen)

【题目描述】

定义 $f(x, y) = (x \oplus y) \times (x | y) \times (x \& y)$, 其中 $\oplus, |, \&$ 分别代表按位异或、按位或、按位与。

给你一张 n 个点, m 条边无向图, 其中无自环, 可能有重边。记点 i 的度数为 deg_i 。

请你求出 $\sum_{i=1}^n \sum_{j=1}^n f(deg_i, deg_j)$, 由于答案可能很大, 输出时请将答案对 $10^9 + 7$ 取模。

【输入格式】

第一行两个正整数 n, m 表示无向图的点数和边数。

接下来 m 行每行两个正整数 u, v 表示一条连结点 u 和 v 的无向边。

【输出格式】

仅一行一个整数表示答案。

【输入输出样例1】

gen.in	gen.out
5 6 1 2 1 3 1 4 5 2 2 5 4 3	36

【数据规模与约定】

对于所有的测试数据, $1 \leq n \leq 2 \times 10^5, 0 \leq m \leq 2 \times 10^6$, 保证给出的无向图无环。

子任务编号	特殊性质	分值
1	$n \leq 2000$	19
2	$\forall i, deg_i \leq 2000$	31
3	无	50

聚众斗殴(fight)

【题目描述】

给定长度为 $n - 1$ 的数组 a 以及一个数字 x 。

将 x 加入数组 a 的某个位置，这样数组中共有 n 个元素。接下来进行如下操作：

1. 取出数组的前两个数 a_1, a_2 。
2. 有 $\frac{a_1}{a_1+a_2}$ 的概率 a_1 获胜， $\frac{a_2}{a_1+a_2}$ 的概率 a_2 获胜。
3. 将获胜的数字移到数组末尾，并删去另一个数字。
4. 如果数组只剩下一个数字，结束该过程，否则重复进行这些操作。

$\forall 1 \leq i \leq n$ ，你需要回答，如果将 x 插入在 a_i 之前（特别地，若 $i = n$ ，将 x 插入到数组末尾）， x 为最后剩下的数的概率。

【输入格式】

第一行输入两个数 n, x 。第二行包含 $n - 1$ 个整数 $a_1, a_2, \cdots, a_{n-1}$ 。

【输出格式】

输出 n 行，第 i 行一个数表示 x 插在 a_i 之前时的答案。

输出的答案和准确答案的相对误差和绝对误差在 10^{-9} 以内都被认为是正确的。

【输入输出样例1】

fight.in	fight.out
4 3	0.188265306122449
4 5 2	0.188265306122449
	0.239285714285714
	0.239285714285714

【数据规模与约定】

对于所有的测试数据，保证 $2 \leq n \leq 4096, n = 2^k, k \in \mathbb{Z}^+, 1 \leq x, a_i \leq 10^4$

子任务编号	子任务分值	$n \leq$	特殊限制
1	15	8	无
2	15	4096	a_i, x 全部相同
3	20	4096	$a_i \leq 10$
4	20	128	无
5	30	4096	无

可怜 (kel)

【题目描述】

给定 n ，你需要给出一个长度为 n 的『kel 序列』 a 。

我们把你个长度为 n 的序列 a 称为『kel 序列』,当且仅当它满足 $\forall 1 \leq l < r < l' < r' \leq n, a_{l\dots r} \neq a_{l'\dots r'}$ 。

Shinomiya 想要你给出一个『kel 序列』，若无解，请输出 Shinomiya。

值得注意的是，『kel 序列』仅由 $[L, R]$ 间的整数组成。

SPJ 会判断你的『kel 序列』并且给出相应分数。

- 如果你给出的『kel 序列』最优，你会获得这个测试点的全部分数（即 5 分）。
- 如果你给出的『kel 序列』合法但非最优，你会获得这个测试点的部分分（即 3 分）
- 如果你给出的『kel 序列』非法，你会在该测试点获得 0 分。

『kel 序列』最优当且仅当其字典序在长度为 n 的『kel 序列』中最小。

【输入格式】

输出 n 行 n 个整数或一个字符串，表示符合要求的『kel 序列』或 Shinomiya。

【输出格式】

输出一行一个长度为 n 的字符串，表示最终得到的字符串。

【输入输出样例1】

kel.in	kel.out
1 0 0	0

【输入输出样例2】

kel.in	kel.out
114514 172003 172300	Shinomiya

【数据规模与约定】

- 对于10%的数据，保证 $1 \leq n \leq 10$ 。
- 对于20%的数据，保证 $1 \leq n \leq 50$ 。
- 对于60%的数据，保证 $1 \leq n \leq 10^3$ 。
- 对于100%的数据，保证 $1 \leq n \leq 10^6, 0 \leq L \leq R \leq 10^9$ 。