

NOIP2023模拟赛

题目名称	计算几何	顶峰远眺	游戏	树上连通问题
题目类型	传统型	传统型	传统型	传统型
可执行文件名	imp	mountain	game	tree
输入文件名	imp.in	mountain.in	game	tree.in
输出文件名	imp.out	mountain.out	game.out	tree.out
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	1.0 秒
内存限	512 MiB	512 MiB	512 MiB	512 MiB
子任务/测试点数目	5	10	20	20
是否等分	否	是	是	是

提交源文件程序名

对于C++语言	imp.cpp	catus.cpp	game.cpp	reborn.cpp
---------	---------	-----------	----------	------------

编译选项

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

注意事项(请仔细阅读)

- 文件名（程序名和输入输出文件名）必须使用英文小写。
- 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。
- 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**。
- 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）。
- 选手提交的程序源文件必须不大于100KB。
- 程序可使用的栈空间内存限制于题目的内存限制一直。
- 使用std::deque等STL容器时，请注意其内存空间消耗。
- 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。
- 评测在Windows 10下进行，使用LemonLine进行评测。

计算几何 (imp)

【题目描述】

在平面直角坐标系上有 n 个点，第 i 个点的坐标为 (x_i, y_i) ，权值为 c_i 。

你需要选择一个四边均平行于坐标轴的正方形（可以退化成一个点），使得该正方形的左下角和右上角都在直线 $y = x$ 上，使得在正方形内（含边界）的点的点权和减去正方形的边长最大。

请你求出这个最大值。

【输入格式】

第一行一个正整数 n 。

接下来 n 行，每行三个正整数 x_i, y_i, c_i 。

【输出格式】

仅一行一个整数表示答案。

【输入输出样例1】

imp.in	imp.out
6 2 3 4 1 4 -4 0 0 2 1 0 -5 3 1 -1 1 1	4

【输入输出样例1说明】

选择左下角坐标为 $(1, 1)$ ，右上角坐标为 $(3, 3)$ 的正方形，此时包含了第 1, 5, 6 个点，边长为 2，故答案为 $4 + (-1) + 3 - 2 = 4$ 。

可以证明，不存在更大的答案。

【数据规模与约定】

对于所有测试数据，满足 $1 \leq n \leq 2.5 \times 10^5, 0 \leq x_i, y_i \leq 10^9, -10^9 \leq c_i \leq 10^9$ 。

子任务编号	特殊性质	分值
1	$n, x_i, y_i \leq 250$	6
2	$n \leq 250$	10
3	$n \leq 2500$	22
4	$n \leq 25000$	24

子任务编号	特殊性质	分值
5	无	38

顶峰远眺 (mountain)

【题目描述】

Farmer John, 也就是 FJ, 带领了他的奶牛们登上了一座座山峰.....

我们抽象地描述山的形状, 有 n 座山峰, 从左到右的高度依次为 h_i , 现在 FJ 想知道, 对于 i , 假设他在山 i 上, 能看到多少山峰?

- 对于 $i, j (i \neq j)$, 我们称 i 可以看见 j 当且仅当对于它们中间所有山 (包括 i, j) 的高度要小于等于 i 的高度。

【输入格式】

第一行, 一个数 n 。

第二行, n 个数, 表示每个山峰的高度 h_i 。

【输出格式】

输出 n 行, 表示对于每个 i 的答案。

【输入输出样例1】

mountain.in	catus.out
5	0
1	1
2	4
3	1
2	0
1	

【数据规模与约定】

对于所有的测试数据, 保证 $1 \leq n, h_i \leq 10^6$ 。

测试点编号	$n \leq$
1 ~ 3	10^3
4 ~ 5	10^5
6 ~ 10	10^6

游戏 (game)

【题目描述】

共有 n 张两面的卡牌，按照 $1 \sim n$ 的顺序排列。卡牌一面黑一面白，初始情况下一些卡牌黑面向上（用 0 表示），剩下的白面向上（用 1 表示）。

小 C 和小 D 轮流操作，小 C 先进行操作。每次每个人必须选择连续的 k 张牌，将这些牌全部翻转为黑面朝上或全部翻转为白面朝上。例如，当 $n = 5, k = 3$ 时，假如目前卡牌的状态是 01010，那么一个人可以将卡牌状态变为 11110、00010、01110 等，但不能变成 10110 或 01010。

如果一次操作之后，所有卡牌都变成黑面朝上或都变成白面朝上，那么进行这个操作的人获胜。

小 C 和小 D 都绝顶聪明，他们都使用最优策略，即尽量让自己获胜、如无法获胜则 尽量不让对方获胜。

请你判断游戏的结果。

【输入格式】

第一行一个正整数 T ，表示测试数据组数。

接下来对于每组数据，第一行两个正整数 n, k ，第二行一个长度为 n 的 01 串，表示初始卡牌状态。

【输出格式】

输出 T 行，每组数据输出一行，对于每组数据，如果小 C 会赢，输出 `win`；如果小 D 会赢，输出 `lose`；否则输出 `tie`。

【输入输出样例1】

game.in	game.out
3	
4 2	
0101	lose
4 1	tie
0101	win
4 2	
0011	

【数据规模与约定】

对于所有的测试数据，保证 $1 \leq T \leq 40, 1 \leq k \leq n \leq 10^5$ 。

测试点编号	n	k	特殊限制
1 ~ 5	≤ 10	≤ 10	无
6 ~ 8	$\leq 10^3$	≤ 2	无
9 ~ 13	$\leq 10^3$	$\leq 10^3$	无
14 ~ 16	$\leq 10^5$	$\leq 10^5$	保证小 D 不会获胜

测试点编号	n	k	特殊限制
17 ~ 20	$\leq 10^5$	$\leq 10^5$	无

树上联通问题 (tree)

【题目描述】

给定一棵 n 个节点的树，和一个正整数 k 。

定义函数 $f(l, r)$ 为，只保留编号在 $[l, r]$ 范围的点时，这个图的连通块数。

关于边的保留：如果一条边 (u, v) 的 u, v 两点同时保留则这条边也一起保留，否则消失。

请对于所有满足 $1 \leq l \leq r \leq n$ 的正整数对 (l, r) ，求出 $f^k(l, r)$ （相当于 $[f(l, r)]^k$ ）之和，答案对 $10^9 + 7$ 取模。

【输入格式】

第一行，两个数 n, k 。

接下来 $n - 1$ 行，每行两个数表示一条边的两个节点。

【输出格式】

输出一行一个数表示答案对 $10^9 + 7$ 取模的值。

【输入输出样例1】

tree.in	tree.out
4 2 1 2 2 3 2 4	13

【输入输出样例1说明】

$f(1, 1) = 1, f(1, 2) = 1, f(1, 3) = 1, f(1, 4) = 1。$

$f(2, 2) = 1, f(2, 3) = 1, f(2, 4) = 1。$

$f(3, 3) = 1, f(3, 4) = 2。$

$f(4, 4) = 1。$

所以答案为 $9 \times 1^2 + 1 \times 2^2 = 13。$

【数据规模与约定】

对于 100% 的数据，保证 $1 \leq n \leq 392699, 1 \leq k \leq 2, 1 \leq u, v \leq n。$

测试点编号	n	k	特殊限制
1 ~ 3	200	2	无
4 ~ 7	5000	2	无
8 ~ 11	392699	1	无

测试点编号	$n \leq$	$k =$	特殊限制
12 ~ 15	392699	2	存在一个节点的度数为 $n - 1$
16 ~ 20	392699	2	无

数据存在一定梯度。