

NOIP2024 模拟赛

题目名称	机器人	可爱捏	总力战	树
题目类型	传统型	传统型	传统型	传统型
可执行文件名	robot	lovely	raid	tree
输入文件名	robot.in	lovely.in	raid.in	tree.in
输出文件名	robot.out	lovely.out	raid.out	tree.out
每个测试点时限	1.0 秒	1.0 秒	2.5 秒	2.0 秒
内存限制	512 MB	512 MB	512 MB	512 MB
子任务/测试点数目	10	20	3	20
是否等分	是	是	否	是

提交源文件程序名

对于C++语言	robot.cpp	lovely.cpp	raid.cpp	tree.cpp
---------	-----------	------------	----------	----------

编译选项

对于C++语言	-O2 -std=c++14
---------	----------------

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。

机器人 (robot)

【题目描述】

你正在研发一款新的机器人，现在要在一块空地上对它进行调试。

空地可以视作一个无限大的二维平面，初始，机器人位于点 $(0, 0)$ 。接下来，它会依次接收 n 条指令，每条指令可以用一个大写的英文字母表示：

- L ：令当前位置为 (x, y) ，机器人移动到 $(x - 1, y)$ 。
- R ：令当前位置为 (x, y) ，机器人移动到 $(x + 1, y)$ 。
- D ：令当前位置为 (x, y) ，机器人移动到 $(x, y - 1)$ 。
- U ：令当前位置为 (x, y) ，机器人移动到 $(x, y + 1)$ 。

需要注意的是，平面上的某些位置可能存在着障碍，当一条指令让机器人向目标位置移动时，会先判定目标位置是否存在障碍。如果存在，机器人会忽略该条指令，并保持原地不动；反之机器人会按照当前指令的方向移动。

你弄丢了所有的实验数据，现在你只知道 $(0, 0)$ 处，即起点不存在障碍，其余障碍的情况以及机器人的最终位置你全部不知道。请设计一个程序来找到执行完 n 条指令后机器人最终可能在的所有位置。

【输入格式】

第一行一个正整数，代表指令条数 n 。

接下来一行是一个只包含字符 L, R, D, U 的长度为 n 的字符串，代表所有的指令。

【输出格式】

首先一行输出一个正整数 k ，代表所有可能的位置数量。

接下来 k 行，每行两个数 (x, y) ，代表一个机器人可能到的位置。特别的，若 $k \geq 2$ ，你需要先将这些坐标按照 x 从小到大排序，对于 x 相同的情况，则按照 y 从小到大排序，最后输出排序后的结果。

【输入输出样例】

robot.in	robot.out
2	4
RU	0 0
	0 1
	1 0
	1 1

【输入输出样例 1 说明】

若 $(1, 0)$ 与 $(0, 1)$ 同时存在障碍物，则机器人会停在 $(0, 0)$ ；若 $(1, 0)$ 存在障碍物，则机器人会停在 $(0, 1)$ ；若 $(1, 1)$ 存在障碍物，则机器人会停在 $(1, 0)$ 。

若不存在障碍物，则机器人会停在 $(1, 1)$ 。

【数据范围与约定】

- 对于前 30% 的测试数据, $n \leq 5$ 。
- 对于所有测试数据, 保证 $1 \leq n \leq 20$ 。

可爱捏 (lovely)

【题目描述】

小 C 有许多可爱的数，当她们两两聚在一起时就更可爱了！不过由于小 C 不希望让她们过分可爱，因此她决定选出一些数使得她们两两乘积不为一个完全立方数QAQ。

即现在小 C 有 n 个整数 $a_i (1 \leq i \leq n)$ ，现在需要选出若干个 a_i 构成一个最大的集合 S ，使 S 中的数两两乘积不为完全立方数。

【输入格式】

第一行输入一个正整数 n 。

第二行 n 个正整数，代表题面中的 a_i 。

【输出格式】

一个整数代表最大集合的大小。

【输入输出样例 1】

lovely.in	lovely.out
2 3 9	1

【输入输出样例 2】

lovely.in	lovely.out
5 27 8 6 36 5	3

【数据规模与约定】

Hint: 完全立方数指的是形如 $a = b^3$ 的数，其中 $a, b \in \mathbb{N}$ ，则称 a 为完全立方数。

- 对于 10% 的测试数据， $n \leq 10^5, a_i \leq 10^6$ 。
- 另有 20% 的测试数据， $n \leq 10^3, a_i \leq 10^{10}$ 。
- 对于 100% 的测试数据， $1 \leq n \leq 10^5, 1 \leq a_i \leq 10^{10}$ 。

总力战(raid)

【题目描述】

C 老师正在指挥一场总力战。

总力战的地图可以抽象成一个二维坐标系，并且有一个宽度值 X 。地图上有 n 个学生，每个学生都有自己的 a_i, b_i, c_i 。对于第 i 个学生，其可以在连接 $(0, a_i)$ 和 (X, b_i) 的线段上自由移动，速度任意，且坐标可以是任意实数。但是，需要注意，任意一个时刻，**不能有两个不同的学生站在相同的位置**。学生个性差异比较大，所以**满足 a_i 互不相同且 b_i 互不相同**。

每个学生的战力会受到其他学生的影响。加入学生 i 的位置是 (x_i, y_i) ，那么学生 u 的战力是所有满足 $x_v = x_u$ 且 $y_v > y_u$ 的学生的 c_v 之和。

C 老师在指挥的过程中，会问你 Q 个问题，用来决定下一步的行动。具体而言，在所有问题之前，他会给你一个参数 $K (K \geq 1)$ 。每个问题，他会给你两个参数 p 和 s 。你需要回答对于学生 p ，加入她的 x 坐标必须处于 $[s, s + K]$ ，但其余学生可以任意移动，那么学生 p 的最大战力是多少。可以证明，这个问题的答案和学生的初始位置无关。

【输入格式】

第一行四个正整数 X, K, n, Q 。

接下来 n 行每行三个正整数 a_i, b_i, c_i 。

接下来 Q 行，每行两个正整数 p, s 。

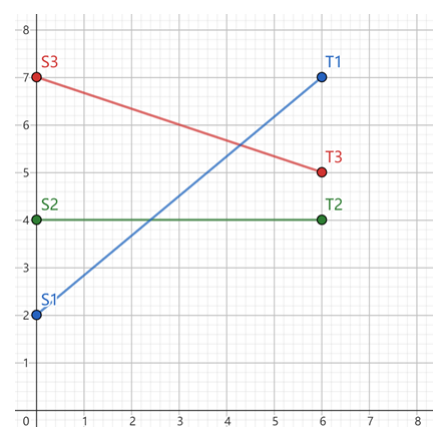
【输出格式】

对于每个问题，输出一行表示答案。

【输入输出样例 1】

raid.in	raid.out
6 2 3 3	
2 7 4	
4 4 2	1
7 5 1	5
1 4	5
2 4	
2 3	

【输入输出样例 1 说明】



对于第一个询问，可以选择 $x_1 = x_3 = 5.14, x_2 = 2.4$ 。

对于第二个询问，可以选择 $x_1 = x_2 = x_3 = 4$ 。

对于第三个询问，可以选择 $x_1 = x_2 = x_3 = \pi$ 。

【数据规模与约定】

对于所有测试数据，保证
 $1 \leq X, a_i, b_i, c_i \leq 10^9, 1 \leq K \leq X, 0 \leq s \leq X - K, 1 \leq p \leq n \leq 2000, 1 \leq Q \leq 8 \times 10^5$ 。

子任务编号	$n \leq$	$Q \leq$	$X \leq$	子任务分值	特殊性质
1	5	5	10	15	不存在三线共点
2	2000	2000	10^9	40	无
3	2000	8×10^5	10^9	45	无

树(tree)

【题目描述】

猪猪又种了一棵 N 个节点的树，初始每条边的长度都为 1，定义树的直径为最短距离最远的两个点的距离。

给定 Q 次询问，每次询问给出 K_i ，你必须正好使用 K_i 次魔法，每次魔法可以任意选择一条边并使它的长度增加 1，你想要让最终树的直径尽可能短，求最短直径。(询问与询问之间相互独立)

【输入格式】

第一行一个正整数 N 。

接下来 $N - 1$ 行，每行两个正整数 a_i, b_i 表示一条边。

接下来一行一个正整数 Q ，表示询问个数。

接下来一行 Q 个正整数 K_1, K_2, \dots, K_Q 表示询问参数。

【输出格式】

输出 Q 行，每行一个整数表示答案。

【输入输出样例 1】

tree.in	tree.out
4	2
1 2	3
1 3	4
1 4	5
10	5
0 1 2 3 4 5 6 7 8 9	6
	6
	7
	8
	8

【输入输出样例 2】

tree.in	tree.out
9	6
1 4	7
2 4	7
3 4	7
4 5	7
5 6	8
6 7	8
7 8	8
8 9	9
10	9
0 1 2 3 4 5 6 7 8 9	9

【数据规模与约定】

- 对于前 15% 的测试数据, $N \leq 10, Q \leq 10$ 。
- 对于前 30% 的测试数据, $N \leq 200, Q \leq 200$ 。
- 对于前 45% 的测试数据, $N \leq 3000, Q \leq 3000, K_i \leq 3 \times 10^6$ 。
- 另有 5% 的测试数据, 树是一条链。
- 另有 5% 的测试数据, 树是一个菊花。
- 另有 20% 的测试数据, 树随机生成。
- 对于 100% 的测试数据,
 $3 \leq N \leq 2 \times 10^5, 1 \leq a_i, b_i \leq N, 1 \leq Q \leq 2 \times 10^5, 0 \leq K_i \leq 10^{18}$ 。