

NOI2020 模拟题

竞赛时间：2019 年 9 月 25 日 8:00—12:00

题目名称	猜测	星际逃亡	圆圈游戏
可执行文件名	guess	rebel	circle
输入文件名	guess.in	rebel.in	circle.in
输出文件名	guess.out	rebel.out	circle.out
每个测试点时限	1 秒	2.5 秒	1.5 秒
内存限制	512MB	512MB	512MB
测试点数目	20	50	25
每个测试点分值	5	2	4
是否有部分分	否	否	否
题目类型	传统型	传统型	传统型

提交源程序须加后缀

对于 Pascal 语言	guess.pas	rebel.pas	circle.pas
对于 C 语言	guess.c	rebel.c	circle.c
对于 C++ 语言	guess.cpp	rebel.cpp	circle.cpp

注意：最终测试时，所有编译命令均不打开任何优化开关。

猜测

【问题描述】

有一块棋盘，棋盘的边长为 100000，行和列的编号为 1 到 100000。棋盘上有 n 个特殊格子，任意两个格子的位置都不相同。

现在小 K 要猜哪些格子是特殊格子。她知道所有格子的横坐标和纵坐标，但并不知道对应关系。换言之，她只有两个数组，一个存下了所有格子的横坐标，另一个存下了所有格子的纵坐标，而且两个数组都打乱了顺序。当然，小 K 猜的 n 个格子的位置也必须都不相同。

请求出一个最大的 k ，使得无论小 K 怎么猜，都能猜对至少 k 个格子的位置。

【输入格式】

输入数据第一行包含一个整数 n 。

接下来 n 行，每行描述一个特殊格子的位置。第 i 行含有两个整数 x_i 和 y_i ，代表第 i 个格子的坐标。保证任意两个格子的坐标都不相同。

【输出格式】

输出一行，包含一个整数，代表最大的 k 。

【样例输入 1】

```
2
1 1
2 2
```

【样例输出 1】

```
0
```

【样例解释 1】

小 K 有可能会猜 $(1,2), (2,1)$ ，此时一个都没对。

【样例输入 2】

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

【样例输出 2】

3

【样例解释 2】

此时只有一种合法的猜测。注意 $(1,1), (1,1), (2,2)$ 不是一个合法的猜测。

【数据规模和约定】

对于 30% 的数据， $n \leq 8$ 。

另外有 5% 的数据，所有横坐标和纵坐标均不相同。

另外有 15% 的数据，所有横坐标或者纵坐标均不相同。

对于 100% 的数据， $n \leq 50$ ， $1 \leq x_i, y_i \leq 100000$ 。

星际逃亡

【问题描述】

你正在逃离银河帝国！然而好景不长，他们已经派出了追兵。你需要小心驾驶飞船，以防被他们发现。

你处于一个有 n 颗星球的星系中。你逃出的地方位于银河帝国的边界，我们记这颗行星的编号为 0 。你在 1 号行星处有一艘穿梭舰，只要逃到了 1 号行星，你就安全了。

你的飞船可以进行空间跳跃，可以在任意两颗行星之间瞬间移动。然而，跳跃系统不稳定，跳跃长度越长，出现危险的可能性越大，因此你想要最小化最长的跳跃距离。

银河帝国有一种特殊的探测仪，可以探测到静止不动的生命物体。如果你在同一颗行星上连续停留超过 S 秒，你就会被发现。因此，你在每颗行星上最多连续停留 S 秒。你可以在任意时刻进行跳跃，而不一定非是整数秒的时刻。在你到达 1 号行星的一瞬，你就会被吸入穿梭舰，因此就不需要再考虑更多的了。

初始时，第 i 颗行星位于宇宙中坐标为 (x_i, y_i, z_i) 的位置，而其运行速度为 (vx_i, vy_i, vz_i) 。这颗行星每秒钟会在三个方向上各移动 (vx_i, vy_i, vz_i) 的距离。注意运动是连续的，而并非每秒钟更新一次的。当然，也有可能静止不动的行星。两颗行星可以在同一时刻处于同一坐标，即便如此，你仍然需要通过空间跳跃在这两颗行星之间移动。

请你求出，在可以成功逃脱的前提下，最长的跳跃距离的最小值是多少？

【输入格式】

输入的第一行包含两个整数 n 和 S ，分别代表行星的颗数和可以保持静止的最长时间。

接下来 n 行，每行包含六个整数 $x_i, y_i, z_i, vx_i, vy_i, vz_i$ ，描述一颗行星。

【输出格式】

输出一行，包含一个实数，代表最长跳跃距离的最小值。只有当你的输出与标准答案的绝对误差不超过 10^{-4} 时，才会被判为正确。

【样例输入 1】

```
3 7
0 0 0 0 0 0
1 2 2 0 0 0
1 1 1 0 0 0
```

【样例输出 1】

```
1.7320508
```

【样例解释 1】

所有行星都是静止的。如果直接跳到 1 号行星上，则最大跳跃距离为 3；如果先跳到 2 号行星，再跳到 1 号行星，则最大跳跃距离为 $\sqrt{3}$ 。

【样例输入 2】

```
5 10
0 0 0 0 0 0
35 0 0 -1 0 0
1 54 0 0 -2 0
2 -150 0 0 10 0
4 0 0 -1 0 0
```

【样例输出 2】

```
2.0000000
```

【样例解释 2】

0 号行星是静止的。我们可以现在 0 号行星等待 4 秒，然后跳到 4 号行星上，等待 1 秒，再跳回 0 号行星。然后我们等待 10 秒，再跳到 3 号行星上，等待 2 秒，再跳到 2 号行星上。接着我们再等待 10 秒，然后跳回 0 号行星，再等待 8 秒，最后跳到 1 号行星上。

整个旅途中最长的跳跃距离是在第 15 秒时从 0 号行星跳到 3 号行星的距离，距离为 2。

【样例输入 3】

```
3 1
-10 2 0 1 0 0
0 0 10 0 0 -1
-10 -2 0 1 0 0
```

【样例输出 3】

```
4.0000000
```

【样例解释 3】

因为 0 号和 2 号行星的速度相同，因此它们始终保持相对静止。我们可以在 0 号和 2 号行星之间来回跳跃，等 1 号行星靠近之后再跳到 1 号行星上。这一方案的最长跳跃距离最小可以达到 4。

【数据规模和约定】

对于 8% 的数据， $n = 2$ 。

对于 20% 的数据， $n \leq 5$ 。

另外有 40% 的数据， $vx_i = vy_i = vz_i = 0$ 。

对于 100% 的数据， $2 \leq n \leq 1000$ ， $1 \leq S \leq 100$ ， $-500 \leq x_i, y_i, z_i, vx_i, vy_i, vz_i \leq 500$ 。

圆圈游戏

【问题描述】

在无聊的时候，小 K 和小 H 会在纸上玩这样一个游戏。

我们可以将纸看做一个平面直角坐标系。小 H 会先在上面画出 n 个圆，并把每个圆的圆心以及半径都告诉小 K。小 H 画的 n 个圆中，任意两个圆不会出现相交或相切的情况。小 K 需要做的就是从这 n 个圆中选出若干个圆，使得选出的任意一个圆都不被另一个选出的圆包含。游戏的目标就是要选出尽量多的圆。

游戏一次一次进行着，小 K 已经对游戏的规则感到了厌倦，所以他决定修改游戏的规则。对于第 i 个圆，我们定义它的价值为 w_i 。新的游戏目标是使得选出的圆价值和最大（不一定数量最多）。但是圆圈可能很多，或者圆圈的分布非常奇怪，或者小 K 还有别的事情要做。所以他只好拜托你来帮他求出这个最大值了。

【输入格式】

输入文件的第一行一个整数 n 表示圆圈的个数。

接下来 n 行每行 4 个整数 x_i 、 y_i 、 r_i 和 w_i ，分别表示第 i 个圆的圆心横坐标、纵坐标、半径，和价值。

【输出格式】

输出一行，包含一个整数，代表选出的圆的最大价值。

【样例输入】

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

【样例输出】

```
7
```

【样例解释】

选择圆 1 和圆 3，价值和可以达到 $3+4=7$ 。

【数据规模和约定】

所有测试数据的范围和特点如下表所示（见下页）：

测试点编号	n 的规模
1	$n = 1$
2	$n = 2$
3	$n = 3$
4	$n = 4$
5	$n = 8$
6	$n = 12$
7	$n = 16$
8	$n = 1000$
9	
10	$n = 2000$
11	
12	$n = 3000$
13	
14	$n = 5000$
15	
16	$n = 60000$
17	
18	$n = 70000$
19	
20	$n = 80000$
21	
22	$n = 90000$
23	
24	$n = 100000$
25	

对于全部测试数据 $1 \leq x_i, y_i, r_i \leq 10^8$, $1 \leq w_i \leq 1000$ 。保证不存在相交或相切的两个圆。