

ANUMFS: The Malaysian Flight Search

Легенда

Легенда, описанная далее, переформулирована и упрощена переводчиком, чтобы читатель мог лучше понять условие задачи. Оригинальную легенду вы можете прочитать на странице задачи в контексте.

Это интерактивная задача — нужно, взаимодействуя с системой, найти чёрный ящик самолёта. Для поиска у вас есть **M** самолётов. Каждый самолёт характеризуется тремя числами: идентификатором **I**, радиусом сигнала **R**, и стоимостью топлива **C**. Изначально все самолёты стоят в точке $(0, 0)$.

Чёрный ящик должен находиться в какой-то точке заданной области поиска (простой многоугольник со сторонами, параллельными осям координат). Для поиска этой точки, вы можете посылать имеющиеся у вас самолёты на разведку. Если вы посылаете самолёт с характеристиками **(I, R, C)** на разведку в точку **(X, Y)** области поиска, тогда если чёрный ящик находится не дальше, чем в **R** единицах длины от точки **(X, Y)**, датчики самолёта обнаруживают, что чёрный ящик находится в радиусе сигнала. Такая операция стоит для вас $2(X+Y)C$ единиц денег. После разведки самолёт возвращается в точку $(0, 0)$. Один самолёт может быть использован сколько угодно раз.

Обратите внимание, что все расстояния в задаче считаются по манхеттенской метрике. То есть расстояние между точками **(X1, Y1)** и **(X2, Y2)** — это $|X2-X1|+|Y2-Y1|$.

Используя самолёты, вам нужно отыскать чёрный ящик, или сообщить, что область поиска ошибочна, и чёрного ящика в ней нет.

Входные данные

В первой строке каждого теста записано целое число **N** — количество точек на границе области поиска. Далее в **N** строках записаны координаты точек границы области в порядке по границе. Гарантируется, что все отрезки границы области поиска параллельны осям координат.

В следующей строке записано целое число **M** — количество самолётов. В следующих **M** строках записаны целые числа **R** и **C** — характеристики очередного самолёта. Идентификаторы даются самолётам от **0** до **M-1** в порядке следования самолётов во входных данных. Гарантируется, что во входных данных всегда есть самолёт с $R=0$.

Выходные данные

Чтобы дать команду самолёту с идентификатором **I** на разведку точки **X, Y** области поиска, нужно вывести строку **I X Y** (точка **X,Y** обязательно должна принадлежать области поиска). В ответ система автоматически выведет в поток стандартного ввода строку **yes**, если чёрный ящик находится в радиусе сигнала, либо **no**, в противном случае.

Если вы хотите сообщить, что нашли чёрный ящик в точке **(A, B)**, выведите **-1 A B**.

Если вы хотите сообщить, что чёрного ящика в области поиска нет. Выведите числа **-1 -1 -1**.

Не забудьте почистить буфер после использования стандартного вывода. Например, в C++ вы можете использовать следующую функцию:

```
bool SendPlane(int id, int x, int y) {  
    printf("%d %d %d\n", id, x, y);  
    fflush(stdout);  
    char temp[4];  
    scanf("%s", temp);  
    return temp[0]=='y';  
}
```

Ограничения**Область поиска:**

- $1000 \leq N \leq 5000$
- Расстояние от точки (0,0) до любой точки области поиска ≤ 100000 .
- $100000 \leq$ количество целых точек в области поиска ≤ 200000 .
- Область поиска может быть заключена в квадрат $1024*1024$.
- Другие подробности можно посмотреть в части генерация тестов.

Самолёты:

- $40 \leq m \leq 50$
- $0 \leq R \leq 600$
- $100 \leq C \leq 10000$
- Гарантируется, что есть самолёт с **R=0**

Пример взаимодействия с системой

```
in: 4
in: 1 1
in: 1 2
in: 2 2
in: 2 1
in: 2
in: 0 100
in: 1 200
out: 0 1 1
in: no
out: 0 2 2
in: no
out: 0 1 2
in: yes
out: -1 1 2
```

Генерация тестов**Область поиска:**

Генерация области поиска происходит в несколько шагов.

- Изначально возьмём квадрат $1000*1000$, с 1000000 точками внутри.
- Далее случайно выберем прямоугольник $W*N$ ($W < 400$, $N < 400$), разместим его на границе исходного квадрата и удалим все точки которые попали внутрь прямоугольника.
- Повторяем предыдущий шаг, пока не многоугольник не будет содержать от 100000 до 200000 точек.
- После чего повторяем данный процесс ещё несколько раз, в зависимости от некоторого случайного условия.
- В итоге мы получаем многоугольник, граница которого задаётся во входных данных.

Самолёты:

Каждый самолёт генерируются случайно, авторы задачи не хотят разглашать как именно. Гарантируется, что среди всех самолётов, есть самолёт с $R=0$.

Расположение чёрного ящика:

С вероятностью 0.65 чёрный ящик находится внутри области поиска. Всегда, когда чёрный ящик находится все области поиска, он находится близко к ней.

Оценка решения

Если в вашей команде (I, X, Y) , точка (X, Y) за границами области поиска (в точности граница разрешается), вы получите вердикт WA.

Если в вашей команде $(-1 X Y)$, точка (X, Y) за границами области поиска, или чёрного ящика нет в точке (X, Y) , вы получите вердикт WA.

Если вы вывели $(-1 -1 -1)$ но на самом деле чёрный ящик находится в области поиска, вы получите вердикт WA.

Обозначим переменной **COST**, суммарную стоимость всех разведок, количество очков, которое вы получите за тест будет равно:

COST/(количество целочисленных точек в области поиска).

Если в результате ваших команд $COST > 10^9$, или количество выведенных команд больше **500000**, вы получите вердикт WA.