

## Задача. Золотые прямые

C header:            `nice_lines_c.h`  
C++ header:        `nice_lines.h`

Роксет, принцесса-пират, прибыла на секретный остров в Ремейнском архипелаге. Ходят слухи, что там захоронено знаменитое сокровище — *золотые прямые*.

Секретный остров представляет собой квадрат размером  $2 \times 10^{12}$  на  $2 \times 10^{12}$  метров. Любую точку острова можно описать декартовыми координатами. Точка  $(0, 0)$  является центром острова, а оси координат параллельны его сторонам.

На острове захоронены  $N$  *золотых прямых*,  $i$ -я прямая (для  $0 \leq i < N$ ) представляет собой множество точек с вещественными координатами  $(x, y)$ , которые удовлетворяют уравнению  $y = a_i x + b_i$ .

Роксет может использовать специальное устройство, которое называется *прямомер*. По заданной точке  $p$  на острове *прямомер* вычисляет сумму расстояний<sup>1</sup> от точки  $p$  до каждой из  $N$  *золотых прямых*.

К сожалению, количество использований *прямомера* ограничено. Помогите Роксет найти сокровище, используя *прямомер* не слишком много раз.

## Протокол взаимодействия

Участник должен реализовать одну функцию

```
(C)   void solve(int subtask_id, int N);  
(C++) void solve(int subtask_id, int N);
```

Эта функция будет вызвана **ровно один раз** при запуске программы.  $N$  обозначает количество *золотых прямых*, спрятанных на острове.

Эта функция может **не более  $Q_{\max}$  раз** вызывать другую функцию

```
(C)   long double query(long double x, long double y);  
(C++) long double query(long double x, long double y);
```

Участник может вызывать эту функцию лишь с параметрами  $-10^{12} \leq x, y \leq 10^{12}$ .

Она возвращает значение, которое вычислит *прямомер* для точки с координатами  $(x, y)$  — то есть, сумму расстояний от точки  $(x, y)$  до каждой из  $N$  *золотых прямых*. Обратите внимание, что сами *золотые прямые* не будут доступны участнику.

Когда участник определил  $N$  *золотых прямых*, участник должен вызвать функцию

```
(C)   void the_lines_are(int* a, int* b);  
(C++) void the_lines_are(std::vector<int> a, std::vector<int> b);
```

В этой функции  $a[i]$  и  $b[i]$  должны описывать  $i$ -ю *золотую прямую*, для  $0 \leq i < N$ . Участник может вернуть прямые в любом порядке.

## Ограничения

- $1 \leq N \leq 100$

<sup>1</sup>Евклидово расстояние от точки до прямой равно длине кратчайшего отрезка, соединяющего заданную точку с произвольной точкой на прямой.

- $-10\,000 \leq a_i, b_i \leq 10\,000$
- Никакие две прямые не параллельны.

## Система оценки

Для вычисления результата на одном тесте используется следующий алгоритм:

- Пусть  $Q$  означает число вызовов функции `query`.
- Если  $Q > Q_{max}$ , или одна из *золотых прямых* не была найдена корректно, то число баллов за тест равно 0.
- Если  $Q \leq Q_{min}$ , то балл за тест равен 1.
- В противном случае оценка за тест вычисляется как  $1 - 0.7 \cdot \frac{Q - Q_{min}}{Q_{max} - Q_{min}}$ .

Баллы за подзадачу определяются как минимальный балл за один из тестов подзадачи, умноженный на число баллов за подзадачу.

### Подзадача 1 (11 баллов)

- $N = 1$
- $Q_{min} = 10\,000$ ,  $Q_{max} = 10\,000$

### Подзадача 2 (13 баллов)

- $N = 2$
- $Q_{min} = 10\,000$ ,  $Q_{max} = 10\,000$

### Подзадача 3 (7 баллов)

- $N = 3$
- $Q_{min} = 10\,000$ ,  $Q_{max} = 10\,000$

### Подзадача 4 (19 баллов)

- $-500 \leq a_i, b_i \leq 500$
- $Q_{min} = 402$ ,  $Q_{max} = 10\,000$

### Подзадача 5 (23 балла)

- $N \leq 30$
- $Q_{min} = 402$ ,  $Q_{max} = 10\,000$

## Подзадача 6 (27 баллов)

- $Q_{min} = 402$ ,  $Q_{max} = 10\,000$

### Пример

Вызовы жюри	Вызовы участника
<pre>solve( /* subtask_id = */ 1, /* N          = */ 1)</pre>	<pre>query(0, 0) возвращает 0 query(1, 1) возвращает 0 the_lines_are( /* a = */ {1}, /* b = */ {0})</pre>