

## Задача А. Площадь многоугольника

Имя входного файла: `area.in`  
Имя выходного файла: `area.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

На плоскости заданы координаты вершин многоугольника в порядке их обхода. Многоугольник не обязательно выпуклый, но не содержит самопересечений. Требуется найти его площадь.

### Формат входных данных

Сначала записано число  $N$  — количество вершин многоугольника ( $3 \leq N \leq 100$ ), затем  $N$  пар вещественных чисел, задающих координаты его вершин.

### Формат выходных данных

Выведите площадь многоугольника не меньше, чем с 3 знаками после десятичной точки.

### Примеры

area.in	area.out
4 0 0 0 2 4 3.5 4 0	11.0

### Замечание

Если выводить вещественные числа как `print(x)`, то иногда они будут странно отформатированы, например,  $10^{-6} = \frac{1}{1000000}$  будет выведено как `1e-6`.

Поэтому числа с заданной точностью следует переводить в строку так:

```
x = 1.34
```

```
"{: .6f}".format(x) # строка "1.340000"
```

Для того, чтобы избежать проблем с погрешностью вашего ответа, если уловие это позволяет, следует выводить числа с максимально возможной точностью, для типа `float` в Питоне это 16 десятичных знаков:

```
print("{: .16f}".format(x))
```

## Задача В. Внутренние точки

Имя входного файла: `points.in`  
Имя выходного файла: `points.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

На плоскости заданы координаты вершин многоугольника в порядке их обхода. Вершины имеют целые координаты. Многоугольник не обязательно выпуклый. Требуется найти количество точек с целыми координатами, которые лежат внутри многоугольника (не на его границе).

### Формат входных данных

Сначала записано число  $N$  — количество вершин многоугольника ( $3 \leq N \leq 100$ ), затем  $N$  пар целых чисел, задающих координаты его вершин. Все координаты не превосходят  $10^4$  по абсолютному значению.

### Формат выходных данных

Выведите количество точек, лежащих внутри этого многоугольника.

### Примеры

<code>points.in</code>	<code>points.out</code>
3 0 0 1 0 0 3	0
4 0 2 2 2 2 0 0 0	1

## Задача С. Пересечение прямоугольников

Имя входного файла: `rect.in`  
Имя выходного файла: `rect.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

На плоскости задано  $N$  прямоугольников с вершинами в точках с целыми координатами и сторонами, параллельными осям координат. Необходимо найти прямоугольник, являющийся их пересечением.

То, что это прямоугольник, докажите самостоятельно.

### Формат входных данных

В первой строке входного файла указано число  $N$  ( $1 \leq N \leq 1500$ ). В следующих  $N$  строках заданы по 4 целых числа  $x_1, y_1, x_2, y_2$  — сначала координаты левого нижнего угла прямоугольника, потом правого верхнего ( $-10^9 \leq x_1 \leq x_2 \leq 10^9, -10^9 \leq y_1 \leq y_2 \leq 10^9$ ). Обратите внимание, что прямоугольники могут вырождаться в отрезки и даже в точки.

### Формат выходных данных

В единственную строку выходного файла поместите описание искомого прямоугольника в том же формате, в котором заданы прямоугольники во входном файле.

Если пересечение заданных прямоугольников пусто, выведите в выходной файл единственное число -1.

### Примеры

<code>rect.in</code>	<code>rect.out</code>
2 0 0 2 2 1 1 3 3	1 1 2 2

## Задача D. Объединение прямоугольников (версия для Python)

Имя входного файла: `union.in`  
Имя выходного файла: `union.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

На плоскости задано  $N$  прямоугольников с вершинами в точках с целыми координатами и сторонами, параллельными осям координат. Необходимо найти площадь их объединения.

### Формат входных данных

В первой строке входного файла указано число  $N$  ( $1 \leq N \leq 300$ ). В следующих  $N$  строках заданы по 4 целых числа  $x_1, y_1, x_2, y_2$  — сначала координаты левого нижнего угла прямоугольника, потом правого верхнего ( $0 \leq x_1 \leq x_2 \leq 10^9, 0 \leq y_1 \leq y_2 \leq 10^9$ ). Обратите внимание, что прямоугольники могут вырождаться в отрезки и даже в точки.

### Формат выходных данных

В выходной файл выведите единственное число — ответ на задачу.

### Примеры

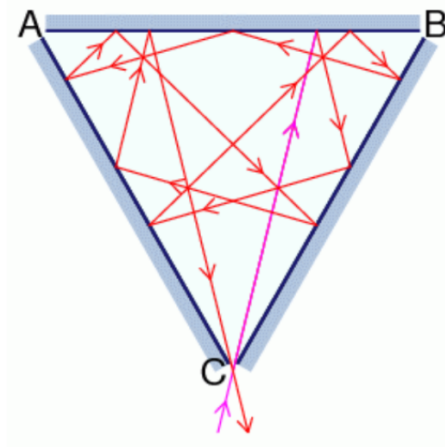
union.in	union.out
3 1 1 3 5 5 2 7 4 2 4 6 7	23

## Задача Е. Зеркальный треугольник

Имя входного файла: `triangle.in`  
Имя выходного файла: `triangle.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Представьте себе равносторонний треугольник, стороны которого зеркальны. Из его вершины выходит луч, и отражается от сторон треугольника, пока не попадает в любую вершину. На рисунке изображен пример для  $K = 11$ . Кроме нарисованного есть еще один способ вернуться в исходную вершину после 11 отражений: пройти по тому же пути обратно, и по два способа попасть в две другие вершины.

Вам нужно для данного  $K$  определить, сколько есть способов попасть в какую-нибудь вершину треугольника, сделав ровно  $K$  отражений.



### Формат входных данных

Одно нечетное число  $K$  ( $1 \leq K \leq 10^5$ ).

### Формат выходных данных

Одно число — количество способов попасть в вершину, сделав ровно  $K$  отражений.

### Примеры

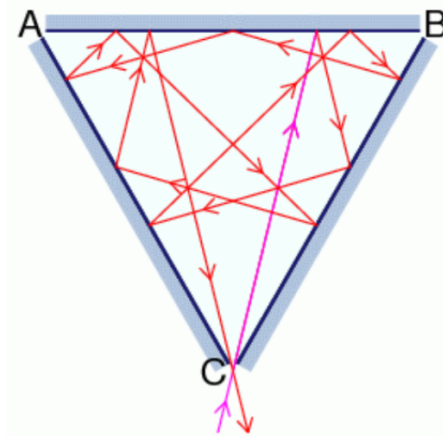
<code>triangle.in</code>	<code>triangle.out</code>
11	6
5	2

## Задача F. Зеркальный треугольник — 2

Имя входного файла: `triangle2.in`  
Имя выходного файла: `triangle2.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Представьте себе равносторонний треугольник, стороны которого зеркальны. Из его вершины выходит луч, и отражается от сторон треугольника, пока не попадает в любую вершину. На рисунке изображен пример для  $K = 11$ . Кроме нарисованного есть еще один способ вернуться в исходную вершину после 11 отражений: пройти по тому же пути обратно, и по два способа попасть в две другие вершины.

Вам нужно для данного  $K$  определить, сколько есть способов попасть в какую-нибудь вершину треугольника, сделав ровно  $K$  отражений.



### Формат входных данных

Одно нечетное число  $K$  ( $1 \leq K \leq 10^9$ ).

### Формат выходных данных

Одно число — количество способов попасть в вершину, сделав ровно  $K$  отражений.

### Примеры

<code>triangle2.in</code>	<code>triangle2.out</code>
11	6
5	2

### Замечание

Задача аналогична задаче "Зеркальный треугольник" но с другими ограничениями