

## Задача А. Векторы

Имя входного файла: `vectors.in`  
Имя выходного файла: `vectors.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайта

Даны два ненулевых вектора. Требуется вычислить:

- Длину первого и второго вектора (два числа)
- Вектор, образованный сложением данных двух векторов
- Скалярное и векторное произведения данных векторов
- Площадь треугольника, построенного из этих векторов

### Формат входных данных

В двух строках входного файла заданы по четыре целых числа, не превосходящих по модулю 10 000, — координаты начала и конца первого вектора, затем второго.

### Формат выходных данных

В каждой строке выходного файла — ответ на соответствующий пункт задачи с точностью не менее  $10^{-6}$ .

### Примеры

<code>vectors.in</code>	<code>vectors.out</code>
5 1 2 6	5.830951895 9.219544457
1 1 7 8	3.000000000 12.000000000
	17.000000000 -51.000000000
	25.500000000

## Задача В. Полярный угол

Имя входного файла: `angle1.in`  
Имя выходного файла: `angle1.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

### Формат входных данных

Два числа — координаты точки ( $|x|, |y| \leq 10^4$ ), не совпадающей с началом координат.

### Формат выходных данных

Одно число — величина её полярного угла в радианах из интервала  $[0, 2\pi)$ . Все числа следует выводить с точностью не менее 6 знаков после запятой.

### Примеры

<code>angle1.in</code>	<code>angle1.out</code>
2 3	0.98279372324732906796

## Задача С. Пусти козла в огород - 1

Имя входного файла: `goat1.in`  
Имя выходного файла: `goat1.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Однажды на огород к Ивану Петровичу забежало целых три козла. Известно, что козлы весьма агрессивные животные, особенно когда речь идет о борьбе за вкусную капусту. Поэтому каждый из трех козлов, заметив других козлов, замер на месте и начал наблюдать за оставшимися козлами: одним глазом за одним козлом, другим — за другим. Естественно, для этого козлу нужно “косить” глазами.

Определите наибольший угол, на который пришлось “раскосить” глазами козлам.

### Формат входных данных

Программа получает на вход координаты трех точки, в которых стоят козлы (сначала координаты первого козла, затем второго, затем третьего). Координаты — пара целых чисел, не превосходящих  $10^4$  по модулю.

### Формат выходных данных

Выведите единственное число — величину самого большого угла, на который “косят” глаза козлов с точностью не меньше 6 знаков после запятой.

### Примеры

<code>goat1.in</code>	<code>goat1.out</code>
0 0	90.000000
3 0	
0 4	

## Задача D. Два треугольника

Имя входного файла: `twotri.in`  
Имя выходного файла: `twotri.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Заданы два треугольника:  $ABC$  и  $DEF$ . Необходимо определить, являются ли они подобными.

### Формат входных данных

Первая строка входного файла содержит координаты вершин треугольника  $ABC$ :  $x_A, y_A, x_B, y_B, x_C, y_C$ . Вторая строка входного файла содержит в аналогичном формате координаты вершин второго треугольника.

Все числа во входном файле целые и не превосходят 5000 по абсолютному значению. Оба описанных во входном файле треугольника имеют ненулевую площадь.

### Формат выходных данных

Выведите в выходной файл слово "YES", если заданные во входном файле треугольники подобны, и слово "NO" — в противном случае.

### Примеры

<code>twotri.in</code>	<code>twotri.out</code>
<code>0 0 1 0 0 1 10 10 12 10 10 12</code>	YES
<code>0 0 1 0 0 1 0 0 10 0 0 5</code>	NO

## Задача Е. Принадлежность точки лучу

Имя входного файла: `point2.in`  
Имя выходного файла: `point2.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

### Формат входных данных

Шесть чисел — координаты точки и координаты начала и конца вектора.

### Формат выходных данных

Одна строка «YES», если точка принадлежит лучу, определяемому вектором, и «NO» в противном случае.

### Примеры

<code>point2.in</code>	<code>point2.out</code>
1 6 3 7 5 8	NO

---

## Задача F. Принадлежность точки отрезку

Имя входного файла: point3.in  
Имя выходного файла: point3.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

### Формат входных данных

Шесть целых чисел от  $-10^4$  до  $10^4$  — координаты точки и координаты концов отрезка.

### Формат выходных данных

Одна строка «YES», если точка принадлежит отрезку, и «NO» в противном случае.

### Примеры

point3.in	point3.out
3 3 1 2 5 4	YES

## Задача G. Пусти козла в огород - 5

Имя входного файла: `goat5.in`  
Имя выходного файла: `goat5.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

В огород к Ивану Петровичу повадился чужой козел по ночам воровать капусту. Чтобы поймать наглеца, Иван Петрович установил на огороде прожектор, освещающий часть плоскости в форме некоторого угла. И когда очередной ночью Иван Петрович услышал хрумку в своем огороде, он включил свой прожектор. Определите, увидит ли Иван Петрович чужого козла или нет.

### Формат входных данных

Программа получает на вход координаты четырех точек  $A$ ,  $O$ ,  $B$ ,  $P$ . Прожектор установлен в точке  $O$ , точки  $A$  и  $B$  лежат на границах освещенной прожектором области (на разных лучах), в точке  $P$  находится козел. Все числа во входном файле целые и не превышают 100 по абсолютному значению. Точки  $A$ ,  $O$  и  $B$  не лежат на одной прямой.

### Формат выходных данных

Выведите слово `YES`, если Иван Петрович увидит козла или слово `NO` в противном случае

### Примеры

<code>goat5.in</code>	<code>goat5.out</code>
0 1 0 0 1 0 1 1	YES
1 0 0 0 0 1 -1 -1	NO

## Задача Н. Пересечение двух отрезков

Имя входного файла: `intersec2.in`  
Имя выходного файла: `intersec2.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Необходимо проверить, пересекаются ли два отрезка.

### Формат входных данных

В единственной строке входного файла даны восемь целых чисел — координаты концов сначала первого, затем второго отрезков. Все числа не превышают 10 000.

### Формат выходных данных

Одна строка «YES», если отрезки имеют общие точки, и «NO» в противном случае.

### Примеры

<code>intersec2.in</code>	<code>intersec2.out</code>
5 1 2 6 1 1 7 8	YES

## Задача I. В каком ухе мяукает Пушок

Имя входного файла:	<code>buzz.in</code>
Имя выходного файла:	<code>buzz.out</code>
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Кирилл знает, что его котик Пушок охотится на новгоднюю ёлочку и хочет её опрокинуть. Кирилл зашел в комнату, в котором находится Пушок и ёлочка, чтобы поймать котика и выгнать его из комнаты. Пушок очень хитёр — он спрятался в комнате, но его выдаёт тихое мяукание. Кирилл находится в точке  $A(x_a; y_a)$  и, глядя прямо на ёлочку, стоящую в точке  $B(x_b; y_b)$  задает вопрос: «В каком ухе у меня раздаётся мяукание?». Пушок спрятался и мяукает в точке  $C(x_c; y_c)$ .

Помогите Кириллу спасти ёлочку — понять, в каком ухе у него раздаётся мяукание.

### Формат входных данных

Программа получает на вход координаты точек  $A$ ,  $B$  и  $C$ . Входные данные являются целыми числами, по модулю не превышающими 1 000.

### Формат выходных данных

Выведите слово `LEFT`, если у Кирилла раздаётся мяукание в левом ухе, `RIGHT` — если в правом, `BOTH` — если мяукание и в левом, и в правом одинаково.

### Примеры

<code>buzz.in</code>	<code>buzz.out</code>
1 0 3 0 0 0	BOTH

## Задача J. Расстояния от точки

Имя входного файла: `distance1.in`  
Имя выходного файла: `distance1.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Даны три точки  $A, B$  и  $C$ . Необходимо подсчитать расстояния от точки  $C$  до прямой, луча и отрезка, образованного точками  $A$  и  $B$ .

### Формат входных данных

В первой строке входного файла даны два целых числа — координаты точки  $A$ . Во двух следующих строках в таком же формате заданы точки  $B$  и  $C$ . Все числа не превосходят по модулю 10 000.

### Формат выходных данных

В первой строке выходного файла выведите одно вещественное число — расстояние от точки  $C$  до прямой. В следующих двух строках выведите соответственно расстояния до луча и до отрезка. Все числа выводить с точностью не менее  $10^{-6}$ . Луч строится по направлению от точки  $A$  к точке  $B$ .

### Примеры

<code>distance1.in</code>	<code>distance1.out</code>
1 1	1.0000000000
2 1	1.0000000000
3 0	1.4142135624

## Задача К. Расстояние между отрезками

Имя входного файла: `distance2.in`  
Имя выходного файла: `distance2.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Найдите расстояние между двумя отрезками.

### Формат входных данных

В двух строках входного файла даны по четыре целых числа, не превосходящих по модулю 10 000, — координаты концов сначала первого, затем второго отрезков.

### Формат выходных данных

В первой строке выходного файла выведите одно вещественное число — расстояние между отрезками с точностью не менее  $10^{-6}$ .

### Примеры

<code>distance2.in</code>	<code>distance2.out</code>
1 1 2 2 2 1 3 0	0.707106781

## Задача L. Пусти козла в огород - 3

Имя входного файла: `goat3.in`  
Имя выходного файла: `goat3.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Козла пустили в огород. К несчастью для козла, его крепко-накрепко привязали к точке  $C$  крепчайшей альпинистской верёвкой длины  $L$  (то есть козёл может кушать травку, удалённую от точки  $C$  не более чем на  $L$  метров). И только одно скрасило жизнь бедного козла — недалеко от него виднеется грядка так любимых им ананасов, представляющая собой отрезок прямой с концами в точках  $A$  и  $B$ . Но, при большом старании, козёл может растянуть крепчайшую альпийскую верёвку. Определите, на какую длину козлу придётся растянуть верёвку, чтобы добраться хотя бы до одного ананасика? А чтобы съесть все ананасы?

### Формат входных данных

В первой строке входного файла содержатся координаты точек  $A$  и  $B$ , разделённые пробелами. На второй строке записаны координаты точки  $C$  и длина верёвки  $L$ , разделённые пробелами. Все числа целые,  $L \geq 0$ , все координаты не превосходят по модулю 10 000.

### Формат выходных данных

В первой строке вывести минимальную длину, на которую козлу придётся растянуть верёвку, чтобы дотянуться до грядки с ананасами. Во второй строке вывести минимальную длину, на которую козлу придётся растянуть верёвку, чтобы добраться до всех ананасов с грядки. Все числа выводить с точностью не менее 6 знаков после запятой.

### Примеры

<code>goat3.in</code>	<code>goat3.out</code>
8 -6 8 6	1.0000000000
0 0 7	3.0000000000

## Задача М. Площадь многоугольника

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

### Формат входных данных

В первой строке одно число  $N$  ( $3 \leq N \leq 100\,000$ ). Далее в  $N$  строках по паре чисел — координаты очередной вершины простого многоугольника в порядке обхода по или против часовой стрелки.

Координаты целые по модулю не превосходят 10 000.

### Формат выходных данных

Одно число — величина площади приведённого многоугольника с абсолютной точностью.

### Примеры

<code>area.in</code>	<code>area.out</code>
3 1 0 0 1 1 1	0.5

## Задача N. Прогулки Сени

Имя входного файла: `walk.in`  
 Имя выходного файла: `walk.out`  
 Ограничение по времени: 2 секунды  
 Ограничение по памяти: 256 мегабайт

Сеня очень любит гулять. Вот и сегодня он пошёл в парк на прогулку.

Сеня решил запомнить свой путь на будущее, вдруг ему на нем понравится какое-нибудь место. Для этого он взял с собой карту, и ставил там точки каждый раз, когда поворачивал или встречал что-то интересное. В результате его пути на карте образовалась ломаная.

Придя домой, он нашёл что один его ботинок стёрся чуть сильнее, чем другой. Сеня предположил, что это из-за того, что поворотов в одну сторону было больше.

Проверьте Сенину догадку, и выведите для данной ломаной, сколько раз она поворачивает налево, а сколько раз — направо.

Учтите, что иногда Сеня возвращается назад или случайно ставит точку там, где поворота не было. Оба таких случая Сеня за поворот не считает.

### Формат входных данных

В первой строке дано число  $N$  ( $3 \leq N \leq 10^6$ ) — количество точек на пути Сени.

Каждая из последующих  $N$  строк содержит координаты точки  $x_i, y_i$  ( $-10\,000 \leq x_i, y_i \leq 10\,000$ ). Две соседние точки не являются равными.

Точки даны в порядке обхода их Сеней.

### Формат выходных данных

Выведите в одной строке два числа: сколько раз Сеня повернул налево, а сколько раз — направо.

### Примеры

walk.in	walk.out
3 0 1 1 0 1 1	1 0
7 0 0 1 1 2 1 3 1 3 3 3 0 0 0	1 2

### Замечание

TL для Python 5 секунд

## Задача О. По компасу

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

Вы, наверное, даже и не догадываетесь, что у пиратов во время принятия новобранца на судно тому приходится проходить тяжёлое испытание. Будущего морского волка высаживают на необитаемый остров, где в определённой точке зарыт клад. Также ему выдаётся компас, с помощью которого можно ориентироваться, а точнее — определить направление на север.

Введём следующие девять типов направлений: обозначим север за  $N$ , юг за  $S$ , запад за  $W$ , восток за  $E$ ; северо-запад за  $NW$ , северо-восток за  $NE$ , юго-запад за  $SW$ , юго-восток за  $SE$ . Если по счастливому стечению обстоятельств новобранец находится ровно над целью, то эта ситуация обозначается буквой  $X$ .

Даны точки  $A$  и  $B$ , задающие соответственно положение новобранца и место, где зарыт клад, и вектор  $C$ , показывающий направление на север. Необходимо определить, к какому из девяти типов, описанных выше, относится направление движения от положения новобранца до клада. Считайте, что направление является северным, южным, западным или восточным только если оно абсолютно точно совпадает с соответствующим вектором. В противном случае относите направление к тому из смешанных типов, между частями которого оно оказалось.

### Формат входных данных

Во входном файле даны координаты точек  $A$ ,  $B$  и координаты вектора  $C$ , разделяемые переводами строки. Все координаты целые и по абсолютной величине не превышают  $10^4$ .

### Формат выходных данных

Выведите строку, обозначающую один из типов направлений, описанных в условии.

### Примеры

<code>treasure.in</code>	<code>treasure.out</code>
0 0 1 1 0 1	NE
1 1 2 2 10 10	N
2 2 3 4 2 -1	W

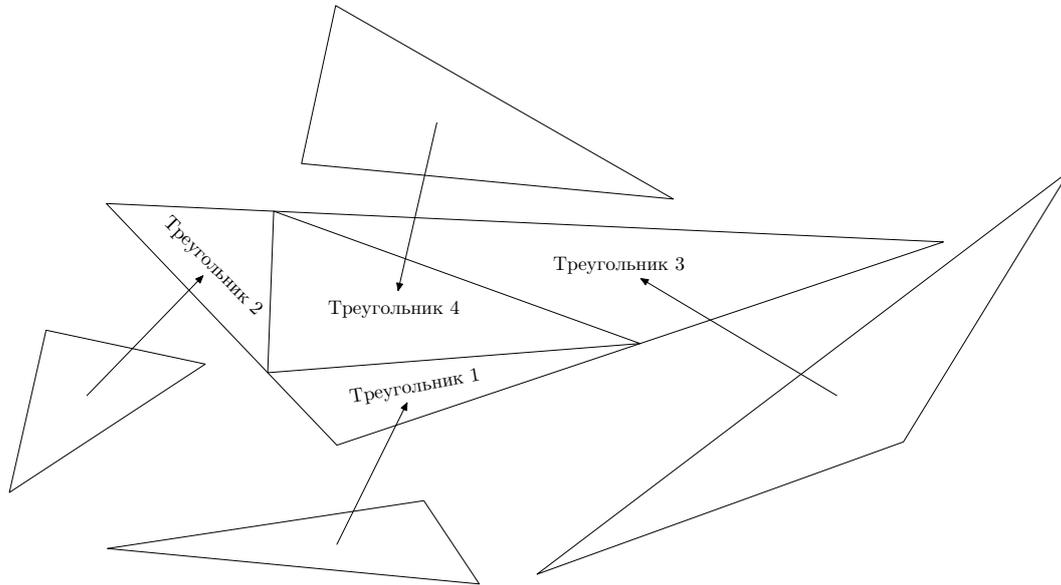
### Замечание

Стороны света при обходе по часовой стрелке идут в следующем порядке: север, восток, юг, запад.

## Задача Р. Треугольная головоломка

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Головоломка состоит из  $n$  треугольников. Чтобы решить головоломку, необходимо выбрать из них четыре треугольника и собрать из них большой треугольник по следующей схеме:



Треугольники не должны пересекаться, в объединении они должны давать треугольник. Ровно по одному из выбранных треугольников должны находиться в углах, а один треугольник должен располагаться в центре.

Треугольники лежат на столе, их можно свободно вращать и двигать, но нельзя зеркально отражать.

Требуется найти все различные наборы из четырех треугольников, из которых можно собрать большой треугольник по указанной схеме. Два набора считаются разными, если существует треугольник, входящий в один, но не входящий в другой.

### Формат входных данных

В первой строке дано одно целое число  $t$  — номер теста.

В второй строке дано одно целое число  $n$  — количество треугольников в головоломке ( $4 \leq n \leq 30$ ).

В следующих  $n$  строках дано описание треугольников. Один треугольник описывается координатами трех своих углов, данных в порядке обхода треугольника против часовой стрелки. Все координаты целые и по модулю не превышают  $10^5$ . Гарантируется, что треугольники не являются вырожденными. В исходном расположении треугольники могут пересекаться.

### Формат выходных данных

В первой строке выведите одно целое число — количество наборов из четырех треугольников, из которых можно собрать большой треугольник по указанной схеме.

В следующих строках выведите наборы. Каждый набор задается номерами треугольников, которые в него входят. Треугольники внутри набора можно выводить в любом порядке. Наборы можно выводить в любом порядке.

**Примеры**

стандартный ввод	стандартный вывод
1 4 0 0 6 2 1 2 0 0 5 0 6 3 0 0 3 1 1 3 0 0 6 3 3 6	1 4 3 2 1
2 6 0 0 1 0 1 1 0 1 0 0 1 0 -1 0 0 0 0 1 1 1 0 1 1 0 -1 0 0 -1 0 0 0 0 1 1 0 1	15 4 3 2 1 5 3 2 1 6 3 2 1 5 4 2 1 6 4 2 1 6 5 2 1 5 4 3 1 6 4 3 1 6 5 3 1 6 5 4 1 5 4 3 2 6 4 3 2 6 5 3 2 6 5 4 2 6 5 4 3

**Замечание**

В первом примере из данных четырех треугольников можно собрать один. При этом треугольники не требуется вращать.

Во втором примере все треугольники имеют одинаковую форму прямоугольного треугольника с длинами катетов равными 1. Из любых четырех треугольников можно собрать один.