

## Задача А. Вписанная окружность

Имя входного файла: `incircle.in`  
Имя выходного файла: `incircle.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 16 мегабайт

Треугольник задан координатами своих вершин. Найдите центр вписанной в него окружности.

### Формат входного файла

Шесть чисел — координаты вершин треугольника..

### Формат выходного файла

Координаты центра вписанной в данный треугольник окружности.

### Пример

<code>incircle.in</code>	<code>incircle.out</code>
0 0 0 15 20 0	5 5

## Задача В. Параллельная прямая

Имя входного файла: `line3.in`  
Имя выходного файла: `line3.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайт

### Формат входного файла

Четыре числа — коэффициенты  $A$ ,  $B$  и  $C$  нормального уравнения прямой и величина  $R$ .

### Формат выходного файла

Шесть чисел — коэффициенты  $A$ ,  $B$  и  $C$  нормального уравнения двух прямых, параллельных заданной и лежащих от неё на расстоянии  $R$  ( $R \neq 0$ ), с точностью до шести знаков после запятой. Порядок прямых не важен.

### Пример

line3.in	line3.out
0 -1 1 1	0 -1 2 0 -1 0

## Задача С. Площадь многоугольника

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

### Формат входного файла

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. Все координаты — целые числа, по модулю не превосходящие  $10^4$ .

### Формат выходного файла

Одно число — величина площади приведённого многоугольника.

### Пример

area.in	area.out
3 1 0 0 1 1 1	0.5

## Задача D. Выпуклая оболочка

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

Дано  $N$  точек на плоскости.

Нужно построить их выпуклую оболочку.

Гарантируется, что выпуклая оболочка не вырождена.

### Формат входного файла

В первой строке число  $N$  ( $3 \leq N \leq 10^5$ ). Следующие  $N$  строк содержат пары целых чисел  $x$  и  $y$  ( $-10^9 \leq x, y \leq 10^9$ ) — точки.

Будьте аккуратны! Точки произвольны. Бывают совпадающие, бывают лежащие на одной прямой в большом количестве.

### Формат выходного файла

В первой строке выведите  $N$  — число вершин выпуклой оболочки. Следующие  $N$  строк должны содержать координаты вершин в порядке обхода. Никакие три подряд идущие точки не должны лежать на одной прямой.

### Пример

hull.in	hull.out
5	4
0 0	0 0
2 0	2 0
0 2	2 2
1 1	0 2
2 2	

## Задача Е. Река

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

Каждый день в Монастыре Светлой Луны проходит утренний обряд, являющийся важнейшим элементом жизненного пути каждого монаха. Во время этого обряда один из монахов должен выйти из Монастыря, пройти к реке Мацанг (истоку Брахмапутры) и, набрав там воды, принести ее в Храм Солнца Дхармы, расположенный неподалеку от Монастыря. При этом обряд должен быть закончен ровно к рассвету. Для того чтобы достичь такой точности, монахам необходимо знать длину кратчайшего пути от Монастыря до Храма, проходящего через реку Мацанг. Вы призваны помочь Монастырю Светлой Луны рассчитать эту самую длину. Поскольку река близ Монастыря не имеет сколько-нибудь значительных изгибов, вы можете считать ее идеально прямой. Кроме того, можно пренебречь ее шириной. Однако необходимо помнить, что Мацанг — река горная и опасная, а поэтому перейти ее вброд абсолютно невозможно! При этом, естественно, ни Монастырь, ни Храм не находятся в реке.

### Формат входного файла

В первой строке записаны координаты Монастыря, во второй — координаты Храма. В третьей строке записаны числа  $k, b$  — коэффициенты, задающие уравнение реки ( $y = kx + b$ ). Все числа кроме  $k$  — целые и не превосходят по модулю 10000. Коэффициент  $k$  — действительное число, не превосходящее по модулю 100.

### Формат выходного файла

Выведите одно действительное число, округленное до трех знаков после запятой — длину кратчайшего пути. В том случае, если Монах не может пройти к Храму, необходимо вывести “No solution.”

### Примеры

river.in	river.out
0 10 20 10 0 0	28.284
0 10 20 -10 0 0	No solution.

## Задача F. Точка в многоугольнике

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

Пускайте, пожалуйста, горизонтальный луч.

Требуется проверить, лежит ли точка внутри или на границе простого многоугольника.

### Формат входного файла

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

### Формат выходного файла

Одна строка «YES», если заданная точка содержится в приведённом многоугольнике или на его границе, и «NO» в противном случае.

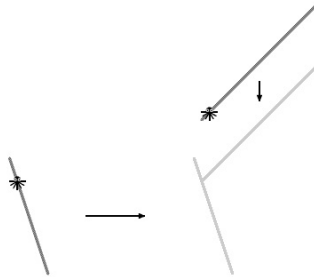
### Пример

<code>point.in</code>	<code>point.out</code>
3 0 0 1 0 0 1 1 1	NO

## Задача G. Про любовь...

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

Паук и паучиха плывут по озеру на двух веточках. Плавать они не умеют, поэтому смогут встретиться только тогда, когда веточки соприкоснутся.



Считая, что веточки имеют форму отрезков, и что они плывут с постоянными скоростями, определите, сколько осталось ждать встречи несчастным членистоногим.

### Формат входного файла

Входной файл содержит 12 чисел:  $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4, v_{1x}, v_{1y}, v_{2x}, v_{2y}$ . Координаты вершин первого отрезка:  $(x_1, y_1)$  и  $(x_2, y_2)$ , координаты вершин второго отрезка:  $(x_3, y_3)$  и  $(x_4, y_4)$ , скорость первого отрезка  $(v_{1x}, v_{1y})$ , скорость второго отрезка  $(v_{2x}, v_{2y})$ . Все числа целые и не превосходят по модулю  $10^4$ . В начальный момент времени веточки не соприкасаются.

Гарантируется, что веточки имеют ненулевую длину.

### Формат выходного файла

Выведите в выходной файл время до ближайшего момента, когда веточки соприкоснутся, с ошибкой не более  $10^{-4}$ . Если веточки не соприкоснутся никогда, выведите число  $-1$ .

### Примеры

<code>love.in</code>	<code>love.out</code>
0 0 -1 3 4 4 7 7 3 0 0 -1	1.6000000000
0 0 -1 3 4 4 7 7 1 0 0 -3	-1

## Задача Н. Выходной

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

Как известно, следующий день после олимпиады является выходным днем. Однако толпы ЛК-Шат стремятся попасть в комповник даже в выходной день. Преподаватели решили запретить ЛК-Шатам приближаться к комповнику. Чтобы ограничить доступ к комповнику жителям определенного домика, проводится линия, разделяющая домик и комповник, которую запрещено пересекать жителям этого домика. Разумеется домик лежит по одну сторону линии, а комповник — по другую.

Будем считать, что как домик, так и комповник представляют собой выпуклые многоугольники на плоскости. Линия, их разделяющая — не что иное, как прямая на плоскости. Многоугольники, соответствующие домику и комповнику, лежат по разные стороны от этой прямой.

Ваша задача — по данным координатам вершин этих многоугольников найти прямую их разделяющую.

### Формат входного файла

В первой строке входного файла задано число  $N$  ( $1 \leq N \leq 50\,000$ ) — количество вершин в многоугольнике, соответствующем домику. Затем в  $N$  строках перечислены координаты вершин многоугольника по часовой стрелке. Затем на отдельной строке задано число  $M$  ( $1 \leq M \leq 50\,000$ ) — количество вершин в многоугольнике, соответствующем комповнику. Затем в  $M$  строках перечислены координаты вершин этого многоугольника по часовой стрелке. Все координаты целые и по модулю не превышают  $10^9$ . В обоих многоугольниках никакие три точки не лежат на одной прямой.

### Формат выходного файла

Выведите в файл три числа  $A, B, C$  — коэффициенты прямой, описываемой уравнением  $Ax + By + C = 0$ . Прямая должна быть нормированной, то есть должно быть выполнено следующее:  $\sqrt{A^2 + B^2} = 1$ . Описанные во входных данных многоугольники должны лежать по разные стороны от выведенной прямой и не должны пересекаться с ней.

### Пример

<code>rest.in</code>	<code>rest.out</code>
4 0 0 0 1 1 1 1 0 3 2 0 2 2 4 0	1 0 -1.5