

## Задача А. Пересечение отрезков

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

Даны два отрезка:  $AB$  и  $CD$ . Определите, какое множество точек является пересечением этих отрезков.

### Формат входного файла

Программа получает на вход восемь целых чисел, по абсолютной величине не превосходящих  $10^4$  — координаты точек  $A, B, C, D$ . Точки могут совпадать (в том числе могут совпадать и концы одного отрезка).

### Формат выходного файла

Если указанные отрезки не пересекаются, то выведите строку «Empty». Если отрезки пересекаются в одной точке, то выведите два числа — координаты точки пересечения. Если пересечением является отрезок, то выведите четыре числа — координаты двух концов отрезка в лексикографическом порядке (то есть сначала нужно вывести ту точку, у которой меньше координата  $x$ , а если у них равны координаты  $x$ , то ту, у которой меньше координата  $y$ ). Все числа следует выводить с точностью не менее 6 знаков после запятой.

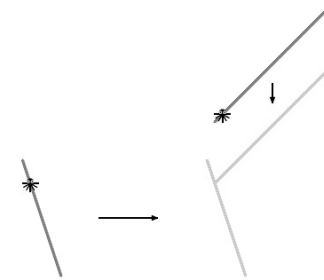
### Примеры

segments.in	segments.out
0 0 9 9 9 5 0 5	5.0000000000 5.0000000000
0 0 9 9 15 15 7 7	7.0000000000 7.0000000000 9.0000000000 9.0000000000
0 0 9 9 10 10 10 10	Empty

## Задача В. Про любовь...

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

Паук и научиха 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

love.in	love.out
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 С. Ловушка для Слонопотама

Имя входного файла: `piglet.in`  
 Имя выходного файла: `piglet.out`  
 Ограничение по времени: 1 секунда  
 Ограничение по памяти: 8 мегабайт

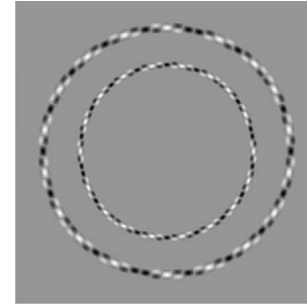
Пятачок и Винни-Пух каждое утро ходят пить чай в гости к Кролику. Естественно,

самым коротким путем.

К сожалению, однажды Винни-Пуху пришла в голову идея вырыть ловушку для Слонопотама. Самое обидное, что они с Пятачком ее даже вырыли. Поэтому теперь каждое утро, идя в гости к Кролику, они боятся в нее провалиться.

Напишите программу, которая посчитает длину самого короткого безопасного пути от домика Винни-Пуха до домика Кролика.

Ловушка для Слонопотама представляет собой яму абсолютно круглой формы. Путь является безопасным, если он не проходит по ловушке (но может проходить по ее границе).



### Формат входного файла

Во входном файле записаны сначала координаты домика Винни-Пуха:  $X_B, Y_B$ , затем — координаты домика Кролика:  $X_R, Y_R$ , а затем — координаты центра и радиус ловушки:  $X_T, Y_T, R_T$ . Все координаты — целые числа из диапазона от  $-32000$  до  $32000$ . Радиус ловушки — натуральное число, не превышающее  $32000$ .

Домики Винни-Пуха и Кролика не могут находиться внутри ловушки, но могут находиться на ее границе.

### Формат выходного файла

Выведите в выходной файл одно число — длину самого короткого безопасного пути от домика Винни-Пуха до домика Кролика с точностью не менее 4 знака после запятой.

### Примеры

piglet.in	piglet.out
0 0 0 1 10 10 1	1.000000
5 0 0 5 0 0 5	7.853982
-5 0 5 0 0 0 3	11.861007

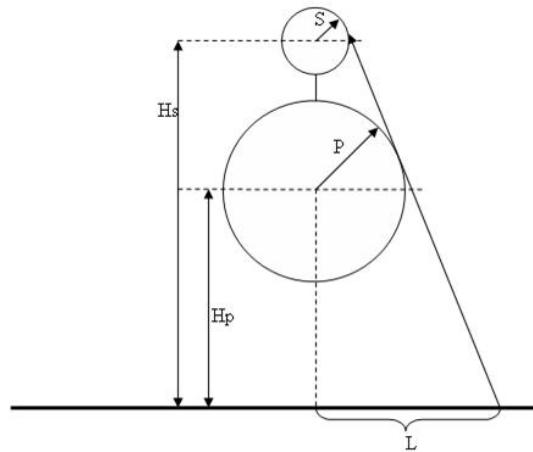
Винни Пух и Пятачок отправились воровать мед у пчел, и, в очередной раз влипли в неприятности. Пятачку опять потребовалось выстрелить из своего охотничьего ружья и пробить воздушный шарик, на котором Винни Пух поднялся к дуплу за медом. При этом желательно попасть именно в шарик, не задев медведя. Вычислите оптимальную позицию для стрельбы.

Поскольку Винни Пух очень любит покушать, то в данной задаче (да и не только в задаче) примем его за сферу радиуса  $R$ . Центр медведя находится на высоте  $H_r$  над уровнем земли. Строго над медведем, находится еще одна сфера, радиуса  $S$  — воздушный шарик; центр шарика находится на высоте  $H_s$  над уровнем земли. Центры обеих сфер находятся на одной вертикальной прямой. По понятным причинам гарантируется, что сферы не пересекаются  $J$ , однако могут касаться.

### Задача D. Собьем воздушный шарик

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

Считая, что ружье стреляет строго по прямой, вычислите минимальное расстояние  $L$ , на которое Пятачок должен отойти от места взлета, чтобы успешно поразить шарик. Шарик считается пораженным, если траектория пули хотя бы касается его поверхности; при этом если траектория пули касается медведя, то он считается невредимым.



### Формат входного файла

В единственной строке входного файла находятся четыре положительных целых числа  $P, H_p, S, H_s$ , не превосходящие 10000.

### Формат выходного файла

Выведите минимальное расстояние от точки взлета, с которого можно поразить шарик из ружья с точностью не менее 5 знаков после запятой.

### Примеры

balloon.in	balloon.out
1 9 10 21	0.0000000

### Задача Е. В школу на велосипеде

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

Петя любит ездить в школу на велосипеде. Но ездить на велосипеде по тротуарам запрещено, а ездить по дороге опасно. Поэтому Петя ездит только по специальным велосипедным дорожкам. К счастью, и Петин дом, и Петина школа находятся в непосредственной близости от таких дорожек.

В городе, где живет Петя есть ровно две велосипедных дорожки. Каждая дорожка имеет форму окружности. В точках их пересечения можно переехать с одной дорожки на другую.

Петя знает точку, в которой он заезжает на дорожку и точку, в которой следует съехать, чтобы попасть в школу. Петю заинтересовал вопрос: какое минимальное расстояние ему следует проехать по дорожкам, чтобы попасть из дома в школу.

### Формат входного файла

Будем считать, что в городе введена прямоугольная декартова система координат.

Первые две строки входного файла описывают велосипедные дорожки. Каждая из них содержит по три целых числа — координаты центра окружности, которую представляет собой соответствующая дорожка, и ее радиус. Координаты и радиус не превышают 300 по абсолютной величине, радиус — положительное число. Гарантируется, что дорожки не совпадают.

Следующие две строки содержат по два вещественных числа — координаты точки, где Петя заезжает на дорожку и точки, в которой Петя съезжает с дорожки. Гарантируется, что каждая из точек с высокой точностью лежит на одной из дорожек (расстояние от точки до центра одной из окружностей отличается от ее радиуса не более чем на  $10^{-8}$ ). Точки могут лежать как на одной дорожке, так и на разных.

### Формат выходного файла

Выведите в выходной файл минимальное расстояние, которое следует проехать Пете по велосипедным дорожкам, чтобы попасть из дома в школу. Ответ должен отличаться от правильного не более чем на  $10^{-4}$ .

Если доехать из дома до школы по велосипедным дорожкам невозможно, выведите в выходной файл число  $-1$ .

### Примеры

bike.in	bike.out
0 0 5 4 0 3 3.0 4.0 1.878679656440357 -2.121320343559643	8.4875540166
0 0 5 4 0 3 4.0 3.0 4.0 -3.0	6.4350110879
0 0 4 10 0 4 4.0 0.0 6.0 0.0	-1

### Note

