

Задача А. 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 В. Пересечение интервалов

Имя входного файла: `segments.in`
Имя выходного файла: `segments.out`
Ограничение по времени: 3 секунды
Ограничение по памяти: 64 мегабайта

Интервал — это отрезок, из которого удалили две точки, являющиеся его концами. Даны два интервала: AB и CD . Определите, какое множество точек является пересечением этих интервалов.

Формат входных данных

Программа должна обработать T наборов входных данных. В первой строке входного файла идёт натуральное число $T \leq 10^5$. В последующих T строках описаны по восемь целых чисел, по абсолютной величине не превосходящих 10^4 — координаты точек A, B, C, D в каждом тесте. Точки могут совпадать (в том числе могут совпадать и концы одного интервала).

Формат выходных данных

Для каждого теста выведите ответ на него в следующей форме:

- Если указанные интервалы не пересекаются, то выведите строку `"Empty"`.
- Если интервалы пересекаются в одной точке, то выведите строку `"Point"` а затем два числа — координаты точки пересечения.
- Если пересечением является интервал, то выведите строку `"Interval"` а затем четыре числа — координаты двух концов интервала 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.

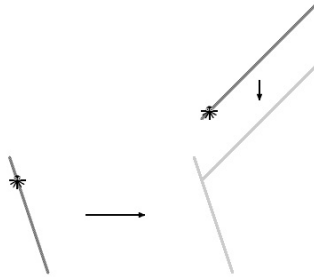
Примеры

<code>segments.in</code>	<code>segments.out</code>
3	Point 5.0 5.0
0 0 9 9 9 5 0 5	Interval 7.0 7.0 9.0 9.0
0 0 9 9 15 15 7 7	Empty
0 0 9 9 10 10 10 10	

Задача С. Про любовь...

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

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 входных данных

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 выходных данных

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 D. 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 входных данных

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 выходных данных

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

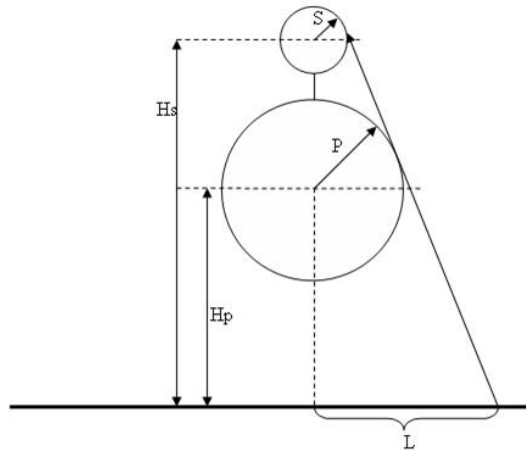
Задача Е. Собьем воздушный шарик

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

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 P . Центр медведя находится на высоте H_p 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, однако могут касаться.

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 входных данных

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

Формат выходных данных

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