

Задача А. Бухгалтерский учёт

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

Жил-был в одной стране Царь по имени Цопа. После очередной царской реформы полномочия Царя стали настолько широкими, что, в частности, он стал собственноручно заниматься финансовой отчётностью.

Известен суммарный доход A его Царства по итогам 0-го года, и суммарный доход B по итогам n -го года (оба числа могут быть отрицательными, что означает — в этот год экономика Царства была убыточной).

Царь хочет продемонстрировать финансовую стабильность, для этого ему надо подобрать единый коэффициент X — коэффициент роста дохода Царства за один год. Этот коэффициент должен удовлетворять уравнению:

$$A \cdot X^n = B.$$

Разумеется, Царь не собирается делать такую работу вручную, и требует от вас написать программу, ищущую этот коэффициент X .

Следует отметить, что дробные числа крайне не любят в экономических структурах Царства, поэтому как входные данные, так и искомый коэффициент должны быть целыми. Искомый коэффициент X может оказаться равным нулю или даже быть отрицательным.

Формат входного файла

В единственной строке записаны три целых числа A , B , n , удовлетворяющих условиям: $|A|, |B| \leq 1000$, $1 \leq n \leq 10$.

Формат выходного файла

Выведите единственное целое число — искомый целый коэффициент X , или фразу «No solution», если такого коэффициента не существует, или он не целый. Если ответов несколько, выведите любой.

Примеры

| <code>stdin</code> | <code>stdout</code> |
|--------------------|---------------------|
| 2 18 2 | 3 |
| -1 8 3 | -2 |
| 0 0 10 | 5 |
| 1 16 5 | No solution |

Задача В. Codeforces World Finals

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

Царю Цопе много раз докладывали о стремительно набирающем популярность сайте Codeforces, на котором собираются светлейшие умы нашего человечества, — ради тренировки и дальнейшего просветления своих умов. И Цопа недавно осознал: если он хочет поработить весь мир, то для этого ему необходимо отдать приказ организовать всемирный турнир Codeforces. Цопа уверен, — после этого его поступка все светлейшие умы добровольно перейдут в его подчинение, и самая сложная часть плана порабощения мира будет осуществлена.

Финальный раунд Codeforces World Finals 20YY назначен на дату *DD.MM.YY*, где *DD* — день проведения раунда, *MM* — месяц, а *YY* — последние 2 цифры года. Васе повезло стать первым в истории Берляндии участником, вышедшим в финал. Но есть одна проблема: по правилам соревнования, все участники на момент финального раунда должны быть не младше 18 лет, а Вася родился в день *BD.BM.BY*. Эта дата написана в Васином паспорте, копию которого он уже отправил организаторам соревнования. Но Вася узнал, что в разных странах форматы записи дат отличаются, например, в США обычно сначала записывается номер месяца, затем день, а потом год. Вася хочет узнать, а можно ли так переставить числа в его дате рождения, чтобы на момент *DD.MM.YY* ему было хотя бы 18 лет. Ведь тогда он сможет сказать, что в его родной Берляндии даты записываются по-другому. Помогите ему справиться с этой задачей.

По еще одному странному правилу, участник должен родиться в том же столетии, в котором проводится финал. Допускается, если участнику исполняется 18 ровно в день финала.

Так как мы рассматриваем годы проведения финала исключительно с 2001 до 2099, то високосным годом будем считать среди них такие, которые делятся на 4.

Формат входного файла

В первой строке задана дата *DD.MM.YY*, во второй строке задана дата *BD.BM.BY*. Гарантируется, что обе даты корректны, и номера годов *YY* и *BY* всегда лежат в отрезке [01;99].

Могло оказаться, что по паспорту Вася родился уже после финала. В этом случае он все равно может менять местами числа в дате.

Формат выходного файла

Если можно переставить числа в дате рождения так, чтобы на момент *DD.MM.YY* Васе было хотя бы 18 лет, выведите YES. Иначе выведите NO.

Каждое число содержит ровно две цифры, и обозначает день, месяц или год в дате. Учтите, что разрешается переставлять любые числа, но не цифры.

Примеры

| <code>stdin</code> | <code>stdout</code> |
|----------------------|---------------------|
| 01.01.98 01.01.80 | YES |
| 20.10.20 10.02.30 | NO |
| 28.02.74 28.02.64 | NO |

Задача С. Тир

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

Однажды в теплый солнечный день Царь Цопа решил сходить в тир, расположенный в Центральном парке, чтобы попробовать выиграть главный приз — большую плюшевую розовую панду. Цопа не очень силен в стрельбе, и поэтому он пригласил Вас помочь ему.

Тир представляет собой бесконечную вертикальную плоскость, на которой введена декартова система координат, на которой появляются мишени. i -ая мишень характеризуется своими координатами x_i и y_i , а также t_i — временем, когда эта мишень появится на плоскости, и числом p_i — вероятностью, с которой Цопа попадает в данную мишень, если он будет целиться в нее.

Мишень появляется и исчезает мгновенно, и Цопа может попасть в нее, если в момент t_i прицел его направлен точно в точку (x_i, y_i) . Скорость перемещения прицела по плоскости равна 1. Все данные заранее известны ему (он же Царь), и поэтому, в частности, он может выбрать, в какую мишень ему целиться с самого начала, в момент времени 0. Цопа действует так, что математическое ожидание количества попаданий максимально.

Формат входного файла

В первой строке задано число n — число мишеней в тире ($1 \leq n \leq 1000$). Далее в следующих строках описаны n мишеней: в i -ой строке идёт описание i -ой мишени. Каждое описание представляет собой четвёрку чисел x_i, y_i, t_i, p_i (где x_i, y_i, t_i — целые, $-1000 \leq x_i, y_i \leq 1000$, $0 \leq t_i \leq 10^9$, а вещественное число p_i задано не более чем с 6 знаками после десятичной точки, и выполняется $0 \leq p_i \leq 1$). Никакие две различные мишени не могут находиться в одной и той же точке.

Формат выходного файла

Выведите единственное число — максимальное математическое ожидание количества попаданий в мишени. Ответ будет принят, если он отличается от правильного не более чем на 10^{-6} .

Примеры

| stdin | stdout |
|-----------------------------|--------------|
| 1 0 0 0 0.5 | 0.5000000000 |
| 2 0 0 0 0.6 5 0 5 0.7 | 1.3000000000 |

Задача D. Задача Царя?

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 3 seconds
Ограничение по памяти: 256 megabytes

Каждый настоящий Царь в своей жизни должен поработить мир, провести финал чемпионата мира Codeforces, выиграть розовую панду в тире и объехать свое царство.

Первые три вещи Царь Цопа уже сделал в своей жизни. Осталось всего лишь объехать свое Царство. Оно, по неизвестным науке причинам, представляет собой бесконечную декартову систему координат, на которой отмечены города. Каждый город будем считать точкой на этой плоскости. Буквально перед самым отъездом, к Царю пришел гонец и сообщил пренеприятнейшее известие о том, что на самом деле города в Царстве расположены в точках с координатами $(x_1, 0), (x_2, 0), \dots, (x_n, 0)$, а еще один город в точке (x_{n+1}, y_{n+1}) (то есть всего их в Царстве ровно $n + 1$).

Но выбора у Царя нет, и он пустился в дальнее путешествие. Начало его пути — это город номер k . Ваша задача — найти такой путь Царя через все города в произвольном порядке (можно даже посещать один город дважды), который имеет наименьшую длину. Закончить путешествие разрешается в любом городе. Можно считать, что между любыми двумя различными городами существует дорога длиной, равной расстоянию между соответствующими точками на плоскости.

Никакие два города не могут находиться в одной точке.

Формат входного файла

В первой строке записаны два целых числа n и k ($1 \leq n \leq 10^5, 1 \leq k \leq n + 1$) — число городов и номер стартового города соответственно. Во второй строке через пробел записано $n + 1$ целых чисел x_i . В третьей строке записано y_{n+1} . Все координаты — целые числа, не превосходящие по модулю 10^6 . Никакие два города не могут находиться в одной точке.

Формат выходного файла

Выведите одно число — наименьшую длину путешествия. Ваш ответ должен иметь относительную или абсолютную погрешность меньше, чем 10^{-6} .

Примеры

| stdin | stdout |
|-------------------------|-------------------------|
| 3 1 0 1 2 1 1 | 3.41421356237309490000 |
| 3 1 1 0 2 1 1 | 3.82842712474619030000 |
| 4 5 0 5 -1 -5 2 3 | 14.24264068711928400000 |

Задача E. Хитрый и умный пароль

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

В эпоху своей молодости герой нашего рассказа Царь Цопа решил, что все его личные данные не очень хорошо скрыты от посторонних глаз, а для монарха это недопустимо. Поэтому он придумал хитрый и умный пароль (лишь спустя много лет Цопа узнал, что его пароль — это палиндром нечетной длины), а затем зашифровал все свои данные.

Как известно, монархи в душе такие же люди, как и все остальные смертные, и поэтому чтобы не забыть свой хитрый и умный пароль, Цопа решил записать его на бумажку. Зная, что опасно хранить пароль в таком виде, он решил его зашифровать следующим образом: с конца и с начала своего пароля-палиндрома он отрезал по x символов (x может быть равно 0, и $2x$ строго меньше, чем длина пароля). У него получились 3 *части пароля*. Назовем их *prefix*, *middle* и *suffix* соответственно, причем *prefix* и *suffix* имеют одинаковую, возможно нулевую, длину, а *middle* — всегда нечетной длины. Из этих трех частей он сделал строку $A + \text{prefix} + B + \text{middle} + C + \text{suffix}$, где A , B и C — некоторые придуманные Цопой строки (возможно нулевой длины), а «+» означает операцию конкатенации («склеивания» строк).

Прошло много лет, и буквально вчера Царь Цопа нашел бумажку, где был записан его пароль, зашифрованный описанным выше способом. Сам пароль, а также строки A , B и C давно уже забыты, и поэтому Цопа просит вас найти пароль максимальной длины, который мог бы быть придуман, зашифрован и записан Цопой.

Формат входного файла

Входные данные представляют собой единственную строку из прописных латинских букв длиной от 1 до 10^5 символов.

Формат выходного файла

В первую строку выведите одно целое число k — количество *непустых частей пароля* в получившемся у вас ответе ($k \in \{1, 3\}$). Далее выведите k строк по два целых числа x_i и l_i — позицию начала и длину соответствующей части пароля. Пары выводите в порядке возрастания x_i . Числа в паре при выводе разделяйте одним пробелом.

Позиция начала x_i — целое число от 1 до длины входной строки. Никакая длина l_i не может быть равна 0, так как части пароля нулевой длины выводить не нужно. Среднее из чисел l_i обязательно должно быть нечетным.

Если решений несколько, выведите любое. Учтите, что ваша задача — максимизировать не k , а сумму l_i .

Примеры

| stdin | stdout |
|---------|------------------------|
| abacaba | 1 1 7 |
| axbya | 3 1 1 2 1 5 1 |
| xabyzba | 3 2 2 4 1 7 2 |