

Задача А. Справка в НацОС

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

НацОС близка к релизу, даже справочная система для неё уже готова! Чтобы ею воспользоваться, нужно нажать кнопку **F1** — это откроет титульную страницу справки. Если же эта страница открыта, то при нажатии на **F1** она закроется. Казалось бы, всё просто. Но и здесь не обошлось без инноваций. . .

Вся справочная информация изложена на нескольких страницах, причём каждая следующая страница описывает всё то же самое, что и предыдущая, но подробнее и более мелким шрифтом. На каждой странице, кроме последней, есть две кнопки: «открыть следующую страницу» и «закрыть следующую страницу». Из всех открытых страниц пользователю видна лишь наименее подробная, а все остальные ему недоступны.

Вы решили научиться пользоваться справкой НацОС. Начните с простого задания: попробуйте изменить набор открытых страниц справки, нажав на кнопки минимальное число раз.

Формат входного файла

В первой строке записано целое число n — количество справочных страниц в НацОС ($1 \leq n \leq 50$). Во второй строке описан начальный набор открытых страниц в виде строки из n нулей и единиц. i -й символ в ней равен единице, если i -я страница открыта, и нулю — в противном случае. В третьей строке в аналогичном формате описан набор открытых страниц, который требуется получить.

Формат выходного файла

Выведите минимальное количество нажатий на кнопки, необходимое для изменения набора открытых страниц.

Пример

<code>standard input</code>	<code>standard output</code>
3 111 000	5

Оптимальная последовательность нажатий: закрыть первую страницу кнопкой **F1**, закрыть третью страницу кнопкой на второй странице, открыть первую страницу кнопкой **F1**, закрыть вторую страницу кнопкой на первой странице, закрыть первую страницу кнопкой **F1**.

Задача В. Урок Цыфиркина

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Феофан, новый ученик Цыфиркина, оказался намного умнее и сообразительнее Митрофанушки. За первые три урока арифметики он уже научился складывать целые положительные числа в столбик — не очень быстро, зато без ошибок. Для этого Феофан использует следующий алгоритм.

- 1) Феофан складывает числа поразрядно справа налево: сначала единицы, потом десятки и т. д.
- 2) Феофан берёт пару цифр в очередном разряде и складывает их.
- 3) Если происходит перенос единицы из предыдущего разряда, Феофан прибавляет её к полученной сумме.
- 4) Он приписывает последнюю цифру суммы к ответу и, если необходимо, делает пометку о переносе единицы в следующий разряд.
- 5) Если разряды закончились, а из последнего разряда есть перенос единицы, Феофан просто приписывает её слева к ответу.

Чтобы написать одну цифру или поставить пометку о переносе единицы, Феофану нужна одна секунда. Если хотя бы одно из складываемых чисел равно нулю или единице, то Феофан тратит на сложение одну секунду, а если оба числа больше единицы — две секунды. Однако, сложив два числа, Феофан запоминает результат сложения и впоследствии в случае надобности вспоминает его за одну секунду. Если Феофану потребуется вычислить $a + b$ и до этого он уже вычислял $b + a$, то он также сможет использовать полученный ранее результат. К сожалению, к следующему уроку Феофан забывает все эти результаты и ему приходится запоминать их заново.

Например, на сложение чисел 526 и 625 Феофан тратит 12 секунд: четыре секунды — на запись цифр ответа, две секунды — на пометки о переносе, две секунды — на вычисление каждой из сумм $6 + 5$ и $2 + 2$ и по одной секунде на вычисление сумм $4 + 1$ и $5 + 6$ (поскольку в процессе сложения Феофан уже запомнил, что $6 + 5 = 11$).

В начале очередного занятия Цыфиркин написал на доске два k -значных числа и дал Феофану задание сложить их. Каждое из этих чисел Цыфиркин выбрал равновероятно из множества положительных k -значных чисел без ведущих нулей. Найдите математическое ожидание времени, которое понадобится Феофану для сложения этих чисел.

Формат входного файла

В единственной строке записано целое число k ($1 \leq k \leq 5000$).

Формат выходного файла

Выведите ожидаемое время сложения двух положительных k -значных чисел с абсолютной или относительной погрешностью не более 10^{-6} .

Пример

standard input	standard output
2	7.51530864

Задача С. Шоу «Ариран»

Имя входного файла:	standard input
Имя выходного файла:	standard output
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

Шоу «Ариран» — это грандиозное представление, проходящее ежегодно в Пхеньяне, столице Северной Кореи. Шоу имеет огромный успех, стоимость билета достигает 300 долларов.

Каждый год в представлении участвуют десятки тысяч артистов. В одном из эпизодов шоу 2011 года n артистов выстраиваются в шеренгу, а потом разбиваются на d хороводов (d — собственный делитель n). В первом хороводе, взявшись по кругу за руки, кружатся артисты с номерами $1, d + 1, 2d + 1, \dots, n - d + 1$, во втором — артисты с номерами $2, d + 2, 2d + 2, \dots, n - d + 2$, и так далее. Последний хоровод образуют артисты с номерами $d, 2d, \dots, n$.

Организаторы шоу попросили вас выбрать цвета рубах, в которые будут одеты артисты, таким образом, чтобы в каждом хороводе любые два соседних артиста были одеты в рубахи разных цветов. Вам сообщили число n , но на вопрос о значении d ответили: «Это число вам знать не положено».

Организаторы могут изготовить рубахи только 26 различных цветов. Сможете ли вы гарантированно выполнить их просьбу вне зависимости от выбора числа d ?

Формат входного файла

В единственной строке записано целое число n ($2 \leq n \leq 300\,000$).

Формат выходного файла

Выведите строку длины n , состоящую из строчных латинских букв. i -й символ должен обозначать цвет рубахи i -го артиста (цвета закодированы буквами от «a» до «z»). Если существуют несколько правильных ответов, выведите любой из них. Если распределить цвета требуемым образом невозможно, выведите «Impossible».

Пример

standard input	standard output
7	acacbdb

Задача D. Надежды гребного слалома

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

15 августа 2008 года Михаил Кузнецов и Дмитрий Ларионов, гребцы из Нижнего Тагила, завоевали бронзовую олимпийскую медаль в соревнованиях каноэ-двоек.

После этого правительство области решило поддержать школу гребного слалома «Полюс» в Нижнем Тагиле, вырастившую спортсменов. В школе на тот момент тренировались n молодых каноистов. Директор школы сообщил, что в годы, предшествующие олимпиаде, m экипажей каноэ-двоек, составленных из учеников школы, занимали призовые места на всероссийских соревнованиях. При этом некоторые каноисты поднимались на пьедестал в составе разных экипажей. Директор попросил выплатить премии спортсменам таким образом, чтобы члены каждого из экипажей-призёров в сумме получили не менее k рублей.

Однако из-за начавшегося экономического кризиса чиновники из министерства физической культуры и спорта решили потратить как можно меньше денег на выполнение просьбы директора школы. Какие премии были выплачены молодым спортсменам?

Формат входного файла

В первой строке записаны целые числа n , k и m ($2 \leq n \leq 500$; $1 \leq k \leq 10\,000$; $0 \leq m \leq 100\,000$). В каждой из следующих m строк записаны два различных целых числа — номера спортсменов из экипажа, ранее занимавшего призовое место. Ученики школы занумерованы целыми числами от 1 до n . Каждый экипаж-призёр встречается в списке ровно один раз.

Формат выходного файла

Выведите в i -й строке размер премии, которую получил i -й спортсмен с абсолютной погрешностью не более 10^{-6} . Если существуют несколько правильных ответов, выведите любой из них.

Пример

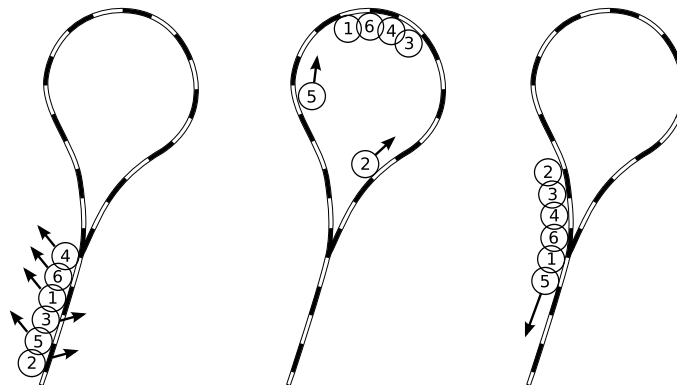
standard input	standard output
4 1000 4	146.5
1 2	853.5
1 3	853.5
2 4	146.5
3 4	

Задача E. Теннисная ракетка

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

После завершения Сталинградской битвы было решено отправить переоснащённые дивизии Красной армии, участвовавшие в операции «Кольцо», на другие участки фронта по железной дороге.

Когда первый поезд с войсками и бронетехникой был сформирован, выяснилось, что железнодорожники не подумали о порядке вагонов. Для сортировки состава решили использовать железнодорожный тупик около химкомбината «Лазурь». Из-за своей формы и, возможно, из-за десятков немецких атак, отбитых на этом участке в ноябре 1942 года, тупик получил у пилотов немецких пикировщиков прозвище «теннисная ракетка».



Состав сортируется следующим образом. Сперва он подгоняется хвостом к развилке, после чего последний вагон отцепляется и заводится в тупик. Все остальные вагоны по очереди отцепляют и заводят в тупик по одному из двух путей, присоединяя его либо к голове, либо к хвосту формируемого в тупике состава. Когда все вагоны заведены в «ракетку», сформированный состав целиком выводят оттуда головой вперёд.

Железнодорожники повторяют эту операцию до тех пор, пока состав не будет отсортирован. Помогите им выполнить эту работу как можно быстрее.

Формат входного файла

В первой строке записано целое число n — количество вагонов в составе ($2 \leq n \leq 10\,000$). В следующей строке записана перестановка чисел от 1 до n — первоначальный порядок вагонов в составе, от головы к хвосту. В результате сортировки вагоны должны быть выстроены по порядку, начиная с первого.

Формат выходного файла

В первой строке выведите целое число k — минимальное количество раз, которое необходимо завести состав в тупик для его сортировки. В каждой из следующих k строк должны быть записаны n чисел — порядок вагонов в составе после очередной операции. Если существуют несколько правильных ответов, выведите любой из них.

Пример

standard input	standard output
6	3
2 5 3 1 6 4	5 1 6 4 3 2
	1 2 3 4 6 5
	1 2 3 4 5 6

Задача F. Болотный доктор

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 3 секунды
Ограничение по памяти: 64 мегабайта

Во время последнего похода к центру Зоны сталкер Штырь угодил в неизвестную ранее аномалию. Дело плохо — у Штыря начали отрастать клыки и покраснели глаза. Теперь от превращения в кровососа Штыря сможет спасти только болотный доктор. . .

Штырь не знает адреса электронной почты доктора, но предполагает, что он есть в КПК погибшего сталкера Семяцкого, который Штырь недавно нашёл. Беда в том, что хитрый Семяцкий не использовал адресную книгу, а прятал адреса электронной почты в содержимом большого текстового файла. Несмотря на то, что Штырь может прочитать весь этот файл, тайну того, какой фрагмент файла является адресом доктора, Семяцкий унёс с собой в могилу.

Штырь решил перебрать все фрагменты текста, которые могут являться адресами, отправляя по одному письму на каждый из них. Сколько же писем ему придётся отправить?

Адрес электронной почты в окрестностях Зоны состоит из имени пользователя и домена, разделённых символом «@». Имя пользователя и домен — непустые строки, состоящие из строчных английских букв и точек. При этом они не могут начинаться на точку, заканчиваться точкой и содержать две точки подряд.

Формат входного файла

Во входных данных записано содержимое файла на КПК Семяцкого. В файле могут встречаться только символы с ASCII-кодами от 32 до 126 и переводы строк. Объём входных данных не превосходит 10^6 байт.

Формат выходного файла

Выведите, сколько различных электронных адресов содержит файл Семяцкого.

Пример

<code>standard input</code>	<code>standard output</code>
<code>a.b@c.d</code> <code>b@c.de.</code> <code>@qq</code> <code>q@.q</code>	5

В файле встречаются следующие адреса: «`a.b@c`», «`a.b@c.d`», «`b@c`», «`b@c.d`», «`b@c.de`».

Задача G. Вавилонская рыбка

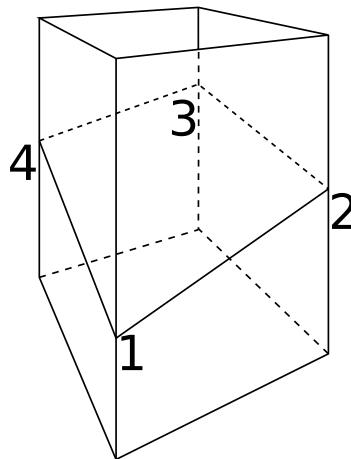
Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Вавилонская рыбка — редкое и очень полезное существо, которое не раз спасало Артура Дента в его путешествиях по галактике. Если вы вставите эту рыбку себе в ухо, вы сможете понимать речь представителей любых рас галактики так, как будто они говорят на вашем родном языке.

После возвращения на Землю Артур достал рыбку из своего уха и решил сбечь её на тот случай, если его друг Форд Префект неожиданно нагрянет и утащит его в очередное путешествие. Артур решил поселить рыбку в аквариуме, который ему подарили разумные дельфины. Тогда он всегда сможет держать аквариум при себе, даже если придётся уехать далеко от дома на автомобиле.

Аквариум представляет собой прямоугольный параллелепипед с квадратным основанием, открытый сверху. Чтобы отслеживать количество воды в аквариуме, Артур установил на каждом вертикальном ребре по датчику, который фиксирует уровень воды на этом ребре.

Состояние просёлочных дорог в Англии далеко от идеального, и из-за того, что машина наклоняется на неровной дороге, поверхность воды может не быть параллельна дну аквариума. В таком случае датчики на рёбрах будут показывать разные значения. Как же тогда узнать объём воды в аквариуме?



Формат входного файла

В первой строке записано целое число t — количество тестов ($1 \leq t \leq 10^4$). В каждой из следующих t строк записаны пять целых чисел: длина стороны квадрата основания аквариума и уровень воды на каждом из вертикальных рёбер. Все эти числа неотрицательные и не превосходят 10^6 . Длина стороны квадрата основания строго положительна. Рёбра описываются в том порядке, в каком они занумерованы на рисунке.

Формат выходного файла

Для каждого теста выведите объём воды в аквариуме, если он определяется однозначно, «`error`», если данные заведомо противоречивы, или «`ambiguous`», если ответ нельзя восстановить однозначно. Все числа нужно выводить с абсолютной или относительной погрешностью не более 10^{-6} .

Пример

standard input	standard output
2	300.000
10 2 3 4 3	error
10 2 3 3 3	

Задача Н. Число Исенбаева

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Владислав Исенбаев — двукратный чемпион Урала по программированию, вице-чемпион TopCoder Open 2009, абсолютный чемпион ACM ICPC 2009. За то время, которое вы потратите на чтение этого условия, Владислав уже решил бы одну задачу. А может, и две. . .

Поскольку Владислав Исенбаев — выпускник СУНЦ УрГУ, неудивительно, что многие из бывших и действующих олимпиадников УрГУ знакомы с ним уже много лет. Некоторые из них с гордостью заявляют, что играли с Владиславом в одной команде. Или играли в команде с бывшими одноклассниками Владислава. . .

Определим *число Исенбаева* следующим образом. У самого Владислава это число равняется нулю. У тех, кто играл с ним в одной команде, оно равняется единице. У тех, кто играл вместе с одноклассниками Владислава, но не играл с ним самим, это число равняется двум, и так далее. Помогите автоматизировать процесс вычисления чисел Исенбаева, чтобы каждый олимпиадник в УрГУ мог знать, насколько близок он к чемпиону ACM ICPC.

Формат входного файла

В первой строке записано целое число n — количество команд ($1 \leq n \leq 100$). В каждой из следующих n строк записаны составы этих команд в виде фамилий трёх участников. Фамилия каждого участника — непустая строка, состоящая из английских букв, длиной не более 20 символов. Первая буква фамилии — заглавная, все остальные — строчные. Фамилия Владислава — «Isenbaev».

Формат выходного файла

Для каждого участника, представленного во входных данных, выведите в отдельной строке через пробел его фамилию и число Исенбаева. Если это число не определено, выведите вместо него «undefined». Участники должны быть упорядочены по фамилии в лексикографическом порядке.

Пример

standard input	standard output
7	Ayzenshteyn 2
Isenbaev Oparin Toropov	Burmistrov 3
Ayzenshteyn Oparin Samsonov	Chevdar 3
Ayzenshteyn Chevdar Samsonov	Cormen undefined
Fominykh Isenbaev Oparin	Dublennykh 2
Dublennykh Fominykh Ivankov	Fominykh 1
Burmistrov Dublennykh Kurpilyanskiy	Isenbaev 0
Cormen Leiserson Rivest	Ivankov 2
	Kurpilyanskiy 3
	Leiserson undefined
	Oparin 1
	Rivest undefined
	Samsonov 2
	Toropov 1

Задача I. Удар самурая

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Коган-сэнсэй продолжает обучать молодого Генноске Фуджики искусству владения мечом. Коган закрепил тонкий стебель бамбука длины l в горизонтальном положении на n опорах и приказал Генноске разрубить его на две части. По словам Когана, меч самурая должен быть настолько острым, а удар должен наноситься так быстро, что после удара обе части стебля должны остаться неподвижными.

Фуджики заметил, что, помимо мастерства самурая, имеет значение и место нанесения удара. Если центр масс одной из частей стебля не будет расположен между двумя опорами, то эта часть упадёт. Генноске предложил покрасить в белый цвет те участки стебля бамбука, после удара мечом по которым хотя бы одна из получившихся частей заведомо упадёт. Помогите ему вычислить суммарную длину участков стебля, которые нужно будет покрасить.

Формат входного файла

В первой строке записаны целые числа l и n ($3 \leq l \leq 10^9$; $2 \leq n \leq 10^5$). Во второй строке записаны n различных целых положительных чисел, меньших l — расстояния от левого конца стебля бамбука до опор, его поддерживающих. Опоры перечислены по порядку, от левого конца стебля к правому. Гарантируется, что изначально стебель расположен устойчиво.

Формат выходного файла

Выведите суммарную длину участков стебля, которые Генноске должен покрасить в белый цвет, округлённую вверх до ближайшего целого числа.

Пример

<code>standard input</code>	<code>standard output</code>
5 4 1 2 3 4	4

Задача J. Ментакулус

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 3 секунды
Ограничение по памяти: 64 мегабайта

Серьёзного человека Ларри, преподавателя квантовой физики в университете Миннесоты, в последнее время преследуют неудачи. Ларри узнал, что жена хочет разорить его, дочь ворует у него деньги на пластическую операцию, у сына проблемы в школе, а руководство факультета получает анонимные письма, очерняющие его.

Но это ещё не всё. Полиция только что арестовала Артура, брата Ларри, несколько месяцев жившего в его доме и круглыми сутками работавшего над загадочной книгой «Ментакулус». Когда Ларри решил почитать «Ментакулус», оказалось, что книга представляла собой собрание рисунков сумасшедшего — все страницы были испещрены причудливыми геометрическими фигурами.

На одной из страниц Ларри увидел множество точек и дуг окружностей. На мгновение ему показалось, что эти точки и дуги образуют ухмыляющиеся рожицы. Неужели в душе Артура не было светлых чувств и вся она была наполнена лишь сарказмом? Ларри решил посчитать, сколько ухмыляющихся рожиц изображено на странице. Для этого, как настоящий учёный, он ввёл строгое определение — дуга окружности PQR и пара точек A и B образуют рожицу, если выполняются следующие условия:

- 1) точки A и B лежат по другую сторону от прямой PR , нежели точка Q ;
- 2) углы APR , ARP , BPR , BRP — острые;
- 3) расстояние от точек A и B до прямой PR меньше удвоенной длины отрезка PR ;
- 4) прямая AB не имеет общих точек с отрезком PR .

Помогите Ларри посчитать, сколько рожиц изображено на рисунке.

Формат входного файла

В первой строке записаны целые числа n и m ($1 \leq n \leq 100$; $1 \leq m \leq 10\,000$) — количество дуг и количество точек соответственно. В каждой из следующих n строк описывается очередная дуга — записаны координаты трёх точек в том порядке, в котором они расположены на дуге. Величины всех дуг не превосходят 180 градусов. Гарантируется, что три точки, задающие дугу, попарно различны и не лежат на одной прямой. В следующих m строках записаны координаты точек. Все координаты целые и не превосходят 10 000 по абсолютному значению. Все дуги и все точки попарно различны.

Формат выходного файла

Выведите количество ухмыляющихся рожиц на странице «Ментакулуса».

Пример

standard input	standard output
2 7 -2 2 0 0 2 2 -2 -2 0 0 2 -2 -1 3 1 3 0 5 0 3 -1 -3 1 -3 2 -3	4

Задача К. Жертва рекламы

Имя входного файла: `standard input`
Имя выходного файла: `standard output`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Фигурист Лев активно готовится к своим пятым Олимпийским играм. В настоящий момент он совместно со своим тренером придумывает новую произвольную программу, которая должна покориť сердца судей и в случае безошибочного исполнения гарантировать Льву золотую олимпийскую медаль.

Тренировки Льва проходят два раза в день, утром и вечером. Его жена, известный продюсер, решила, что время, потраченное мужем на тренировки, можно использовать с пользой для семейного бюджета. Она подписала контракт с рекламным агентством «Увидел — покупай» на съёмки её мужа в рекламе косметики. Рекламное агентство будет снимать тренировки Льва, а потом смонтирует из отснятого материала рекламный ролик.

На очередную тренировку Льва пришли n операторов. Посоветовавшись, они по очереди решили снимать Льва во время выполнения его произвольной программы: сначала съёмку ведёт первый оператор, затем — второй и так далее. Каждый оператор сказал, что идеально будет, если Лев во время съёмки проедет по отрезку прямой из некоторой точки в некоторую другую (каждый оператор указал свою пару точек). Лев решил проехать по всем отрезкам, указанным операторами, переходя с отрезка на отрезок по дуге окружности, так, чтобы его траектория имела вид гладкой кривой. Если не существует дуги, без изломов соединяющей два соседних ориентированных отрезка, то Лев может продлить один из этих отрезков так, чтобы их можно было соединить дугой.

Когда Лев смог построить гладкую кривую, проходящую через все отрезки в указанном порядке, ему стало интересно, за какое минимальное время он сможет проехать по этой кривой. Лев знает, что не может скользить со скоростью больше 10 м/с и с тангенциальным ускорением, превосходящим по модулю 1 м/с² (тангенциальным называется ускорение, направленное вдоль траектории). Кроме того, Лев не может двигаться по дугам окружностей с центростремительным ускорением больше 1 м/с². Напомним, что центростремительное ускорение вычисляется как v^2/R , где v — скорость движения, а R — радиус дуги.

Формат входного файла

В первой строке записано целое число n ($1 \leq n \leq 1000$). В i -й из следующих n строк даны координаты начала и конца ориентированного отрезка, указанного i -м оператором. Координаты целые и не превосходят по модулю 1000. Никакие два соседних отрезка не сонаправлены. Гарантируется, что Лев может построить гладкую кривую, проходящую через все эти отрезки.

Формат выходного файла

Выведите минимальное время (в секундах), за которое Лев сможет проехать по гладкой кривой, проходящей через отрезки, указанные операторами. Лев должен начать движение в начале первого отрезка и закончить его в конце n -го отрезка. Скорость Льва в моменты начала и окончания движения равна нулю. Время должно быть выведено с абсолютной или относительной погрешностью не более 10^{-6} .

Примеры

standard input	standard output
1 0 0 4 0	4.0000000000
2 -2 4 0 4 4 0 4 -2	7.1415926536