

Задача А. 4В и зоопарк

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

4В класс пошел в зоопарк со своей учительницей. К несчастью для неё, детей много, а она одна, поэтому ей сложно за всеми следить. Чтобы упростить себе задачу, она решила разбить детей на группы по 2 или 3 школьника. При этом каждый школьник должен оказаться ровно в одной группе! Задача осложняется тем, что дети не хотят находиться в группах с теми, кого они не знают.

В 4В n учеников. Учительница знает, с кем дружит каждый ученик (ученики дружат в обе стороны!) и что каждый дружит хотя бы с $\lceil \frac{n}{2} \rceil$ учениками, где $\lceil x \rceil$ — минимальное целое число, большее или равное x . Помогите учительнице разбить детей на группы.

Формат входных данных

В первой строке задано целое число n ($2 \leq n \leq 1000$).

Далее идут n строк, i -я из которых содержит через пробел числа $m_{i,1}, m_{i,2}, \dots, m_{i,n}$, каждое из этих чисел — 0 или 1 ($m_{i,j} \in \{0, 1\}$). Если $m_{i,j} = 1$, то это означает, что i -й и j -й школьники дружат, иначе — что они не дружат. Гарантируется, что $m_{i,j} = m_{j,i}$ и что $m_{i,i} = 0$.

Формат выходных данных

В первой строке выведите «YES» (без кавычек), если школьников можно поделить на группы, иначе «NO» (без кавычек). В случае «NO» дальше выводить ничего не нужно.

Во второй строке выведите число k — число групп, на которые можно разбить учеников.

В i -й из следующих k строк выведите a_i — размер i -й группы. Далее в той же строке выведите a_i чисел через пробел — номера учеников в i -й группе. Ученики нумеруются от 1 до n в том порядке, в котором они даны во входных данных.

Примеры

стандартный ввод	стандартный вывод
2 0 1 1 0	YES 1 2 1 2
5 0 1 0 1 1 1 0 1 0 1 0 1 0 1 1 1 0 1 0 1 1 1 1 1 0	YES 2 3 1 2 5 2 3 4

Задача В. Считалочка

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В лагере ребятам надо выбрать капитана команды из n человек, но все хотят эту должность себе. Олег сказал, что он знает интересный способ выбрать капитана.

У каждого участника есть номер на футболке от 0 до 9. Олег придумал натуральное число k , затем выстроил участников в ряд так, чтобы номера на футболках образовывали зацикленную последовательность цифр от 0 до 9: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, 1, 2, 3, 4, ... Затем с помощью считалочки нашёл всех участников на позициях, кратных k , то есть на позициях с номерами $k, 2k, 3k, \dots$ (Первый участник в ряду имеет позицию 1, последний — позицию n). Всех этих участников он построил в новый ряд, сохраняя их относительный порядок, и снова выбрал среди них тех, кто находился в позициях $k, 2k, 3k, \dots$. Выбранных участников построил в новый ряд, и так далее.

Олег повторял это до тех пор, пока в ряду не стало меньше, чем k участников. После чего последнего человека в ряду (то есть человека с наибольшей позицией) он назначил капитаном. Какой номер записан на футболке у капитана?

Формат входных данных

В первой строке задано единственное число n ($1 \leq n \leq 10^9$) — количество человек в команде.

Во второй строке задано единственное число k ($2 \leq k \leq 10^9$) — число, по которому определяются позиции участников, переходящих в следующий ряд.

Формат выходных данных

Выведите одно целое число — номер на футболке у того, кто стал капитаном команды.

Примеры

стандартный ввод	стандартный вывод
19 3	7
123456789 321	2
10 20	9

Задача С. Маджонг

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

С недавнего времени Кейкумусуме начала играть в риичи-маджонг. Несмотря на странные и запутанные правила, последней проблемой осталось быстро определять состояние руки.

В игре каждая кость имеет масть (буквы a , b , c) и номер (число от 1 до 9). Рука состоит из 14 случайно взятых костей и считается *победной* лишь в том случае, когда содержит 4 тройки и одну пару. *Почти победной* рука считается в случае, если для победы необходимо заменить ровно одну кость в руке. Если ни одно из ранее описанных состояний не выполняется, рука считается *несобранной*. При этом любая кость, взятая в пару или тройку, не может входить в любую другую пару или тройку.

Пара — 2 одинаковые кости, например $(1a, 1a)$.

Тройки могут быть двух типов:

- Одинаковые — 3 одинаковые кости, например $(1a, 1a, 1a)$ — корректная одинаковая тройка, а $(1a, 1b, 1c)$ — нет.
- Последовательные — 3 кости одной масти, образующие последовательность с шагом 1, например $(1a, 2a, 3a)$ или $(4c, 5c, 6c)$ — корректная последовательная тройка, а $(1a, 2a, 4a)$ или $(1a, 2b, 3c)$ или $(1c, 3c, 5c)$ — нет.

Упростите ей жизнь и напишите программу, которая будет определять состояние руки за неё.

Формат входных данных

В единственной строке через пробел заданы 14 костей, каждая кость состоит из пары символов (число от 1 до 9 и масть a , b или c). При этом гарантируется, что костей каждого типа не более 4 (однако при почти победной руке после замены разрешается иметь 5 костей одного типа).

Обратите внимание, что при тестировании между 7-й и 8-й костью переноса строки не будет.

Формат выходных данных

Выведите единственную строку — состояние заданной руки, записанное одним из трёх слов: «Tsumo» (победная), «Tenpai» (почти победная) или «Noten» (несобранная).

Примеры

стандартный ввод	стандартный вывод
1a 2a 3a 6a 7a 8a 5b 5b 5b 9c 8c 7c 1c 1c	Tsumo
1a 2a 3a 6a 7a 8a 5b 5b 5b 9c 8c 7c 1c 1b	Tenpai

Замечание

В первом тесте мы можем собрать 3 последовательных тройки $(1a, 2a, 3a)$, $(6a, 7a, 8a)$, $(7c, 8c, 9c)$ и 1 одинаковую тройку $(5b, 5b, 5b)$, а также пару $(1c, 1c)$. В сумме мы получили ровно 4 тройки и одну пару, поэтому рука считается победной.

Во втором примере для победы нам нужно заменить кость $1c$ на $1b$ или наоборот.

Задача D. Таможенные пошлины

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В Древней Руси в период Золотой Орды
тамгой именовалась торговая пошлина.

Марков Л. Н.

Как известно, если вес посылки превышает mxw , получатель должен заплатить по a кредитов за каждый килограмм сверх mxw . Также, если стоимость товара превышает $m xp$ кредитов, получателю необходимо доплатить b процентов от каждого кредита сверх $m xp$.

Недавно в таможеню пришли сотрудники из СКБ Контур и установили новую систему, которая оцифровала информацию о всех посылках и автоматизировала подсчет доплаты. Егор обманом получил доступ к этой системе и теперь намеревается заработать на этом. Он собирается *незначительно изменить данные* не более чем у k посылок среди n лежащих на складе. *Незначительно изменить данные* — изменить ровно одну цифру либо у веса, либо у стоимости товара. Конечно, он хочет, чтобы в результате всех изменений пошлина со всех посылок вышла максимальной. За некоторый процент с прибыли он попросил Вас написать программу, которая решит за него эту задачу.

Формат входных данных

В первой строке через пробел записаны целые числа n и k ($1 \leq n \leq 2 \cdot 10^5$), ($0 \leq k \leq n$) — количество посылок и количество возможных правок.

Во второй строке через пробел записаны целые числа mxw , $m xp$, a , b ($1 \leq mxw, m xp, a \leq 10^9$), ($1 \leq b \leq 100$).

В следующих n строках через пробел записаны целые числа w_i и p_i ($1 \leq w_i, p_i \leq 10^9$) — вес и цена i -й посылки.

Формат выходных данных

Выведите одно действительное число — максимальную пошлинную выгоду от посылок. Ваш ответ будет считаться правильным, если абсолютная или относительная его погрешность не превышает 10^{-9} . Формально, пусть ваш ответ равен x , а ответ жюри равен y . Ваш ответ будет считаться правильным, если $\frac{|x-y|}{\max(1,|y|)} \leq 10^{-9}$.

Далее выведите n строк, в i -й из которых укажите через пробел новый вес и стоимость i -й посылки. Если существует несколько вариантов ответа, выведите любой из них.

Пример

стандартный ввод	стандартный вывод
4 2	3217.000000000
30 500 4 30	6 9100
6 1100	60 90
60 90	9 990
9 990	91 920
91 420	

Задача E. Among Us

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Ходят слухи, что на корабле для межзвёздных путешествий участились саботажы. Для решения этой проблемы компания перевозок разработала специальное сканирующее устройство для определения самозванцев, пытающихся нарушить целостность корабля, среди членов экипажа.

На данном этапе разработки устройство может собирать информацию обо всех членах экипажа и хранить в удобном для себя виде. Этот вид представляет собой единственную строку S , состоящую из слов «Crewmate» (Член экипажа) и «Impostor» (Самозванец), не разделённых пробелами. Остается лишь реализовать модуль, который преобразует эту строку и скажет, сколько самозванцев сейчас находится на корабле. А что произойдет на этом корабле дальше, пускай экипаж решает сам.

Формат входных данных

В первой строке задана строка S ($8 \leq |S| \leq 96$, где $|S|$ — длина строки).

Формат выходных данных

Если на корабле есть самозванцы, выведите единственную строку «There is x Impostor among us», где x — количество самозванцев. В противном случае, выведите единственную строку «Clear».

Примеры

стандартный ввод	стандартный вывод
ImpostorCrewmateCrewmate	There is 1 Impostor among us
CrewmateCrewmate	Clear
ImpostorImpostor	There is 2 Impostor among us

Задача F. Золотые яблоки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

С недавнего времени рынок яблок испытывает новую систему ценообразования. После покупки яблока у продавца цена на следующее яблоко возрастает на 1.

Сегодня на рынке работают Вася и Петя, они продают первое яблоко по цене a и b рублей соответственно. В связи с постоянными поставками можно считать, что яблок у них неограниченное количество.

Узнав о новой системе, Маша собралась на рынок рано утром, чтобы купить N яблок. Заняв первое место в очереди, она задумалась: какое минимальное количество рублей ей понадобится, чтобы купить нужное количество яблок?

Формат входных данных

В первой строке задано целое число N ($1 \leq N \leq 10^9$) — число яблок, которые собирается купить Маша.

Во второй строке задано целое число a ($1 \leq a \leq 10^9$) — цена за первое яблоко у Васи.

В третьей строке задано целое число b ($1 \leq b \leq 10^9$) — цена за первое яблоко у Пети.

Формат выходных данных

В первой строке выведите одно целое число — минимальная сумма, которую потратит Маша на покупку N яблок.

Примеры

стандартный ввод	стандартный вывод
5 4 7	29
6 4 7	37
100 1 1000	5050

Задача G. Идеальный отряд

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

Как известно, идеальный отряд состоит из война, мага, клирика и разбойника. Однако иногда приходится обходиться меньшим: воин, маг и клирик могут образовать отряд из трёх человек, который будет не идеальным. Можно составить отряд и из одного война, если больше некуда деваться.

В вашей гильдии состоит n человек. Каждый человек обладает набором квалификаций: он может быть воином, магом, клириком и/или разбойником. А именно, человек обладает любым непустым подмножеством квалификаций, то есть, он может иметь произвольный набор из одной, двух, трёх или четырёх квалификаций.

Чтобы собрать отряд, нужно взять до четырёх человек и дать каждому роль, соответствующую одной из его квалификаций. На каждую роль можно назначить не более одного человека, и одному человеку нельзя дать две роли сразу, даже если он обладает несколькими квалификациями. Таким образом, в отряде будет не более одного человека в роли война, не более одного человека в роли мага и т.д., и всего будет занято ролей столько, сколько человек в отряде.

Вам нужно собрать один отряд наибольшего размера, и если этот отряд не идеален — разместить объявление о приёме новых людей в гильдию. Чтобы составить объявление, требуется для каждой из четырёх квалификаций выяснить, увеличится ли максимальный возможный размер отряда, если нанять человека только с этой квалификацией.

Формат входных данных

В первой строке дано единственное целое число n — количество людей в гильдии ($1 \leq n \leq 15$).

Каждая из следующих n строк содержит от 1 до 4 заглавных букв W (Воин), M (Маг), C (Клирик), R (Разбойник), не разделённых пробелом, обозначающих набор квалификаций очередного человека. Буквы будут идти именно в таком порядке, то есть каждая из этих строк получается из строки WMCR удалением не более трёх букв.

Формат выходных данных

В первой строке выведите единственное число — наибольший размер отряда, который можно собрать.

Если наибольший размер отряда меньше 4, выведите ещё одну строку, начинающуюся с «Looking for», где далее через пробел перечислены квалификации в объявлении. Каждая квалификация — это одно из четырёх слов: «warrior» (Воин), «mage» (Маг), «cleric» (Клирик) или «rogue» (Разбойник), именно в таком порядке. Указывать нужно такие квалификации, что при найме одного человека с только одной из этих квалификаций наибольший размер отряда увеличится.

Примеры

стандартный ввод	стандартный вывод
1 W	1 Looking for mage cleric rogue
2 M CR	2 Looking for warrior cleric rogue
4 WMCR WMCR WMCR WMCR	4
1 WMCR	1 Looking for warrior mage cleric rogue

Задача Н. Полёт на ядре

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Барон Мюнхгаузен участвует в очередной баталии, которая происходит на плоскости с прямоугольной декартовой системой координат. Командир приказал ему как можно быстрее попасть в центр координат. Дело за малым — нужно сесть на ядро и спрыгнуть с него как можно ближе к цели.

На поле есть n пушечных батарей, и Мюнхгаузен находится в одной из них. Батареи можно считать точками, i -я батарея имеет координаты (x_i, y_i) . Находясь в одной из батарей, барон может выстрелить в сторону любой другой батареи и сесть на ядро; стрелять мимо батареи нельзя — это подорвёт авторитет барона. Ядро не останавливается, достигнув батареи, а продолжает лететь бесконечно далеко. Пролетая над любой из батарей, Мюнхгаузен может спрыгнуть с ядра; после этого он снова сможет выстрелить в сторону другой батареи. Таким образом барон может свободно перемещаться между батареями.

Когда барон решит, что он близок к центру координат, он спрыгнет с ядра и дальше пойдёт пешком. Определите наименьшее расстояние, на которое барон Мюнхгаузен может приблизиться к центру координат, перемещаясь только на ядрах.

Формат входных данных

В первой строке задано целое число n ($2 \leq n \leq 10^5$) — количество батарей.

В следующих n строках заданы целые числа x_i и y_i , разделённые пробелом ($-10^5 \leq x_i, y_i \leq 10^5$) — координаты очередной батареи. Гарантируется, что не существует двух батарей с совпадающими координатами.

Формат выходных данных

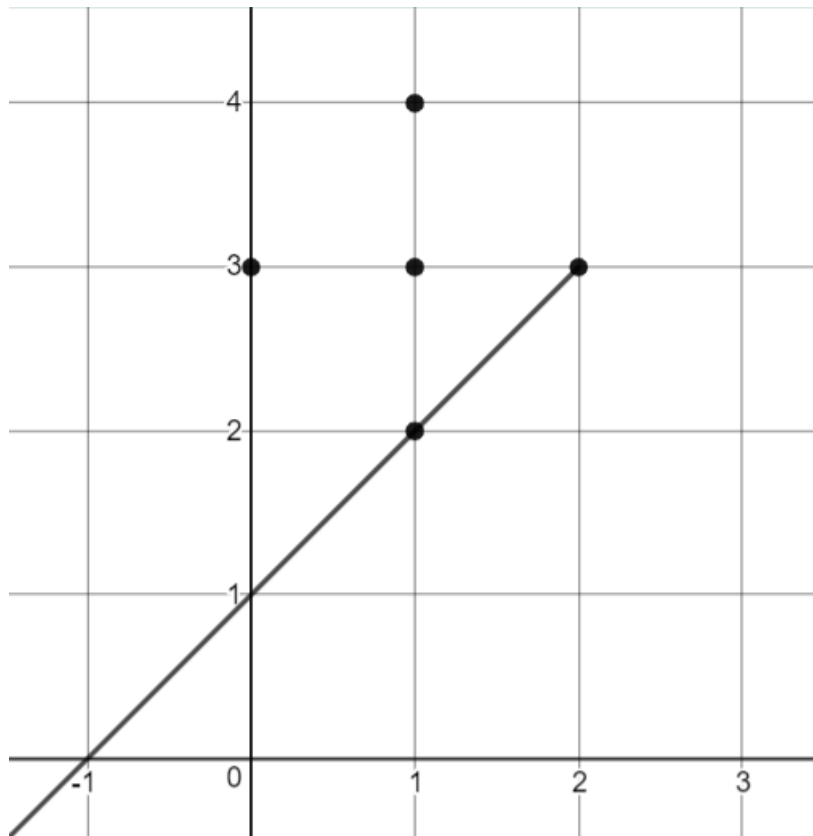
Выведите одно действительное число — наименьшее расстояние, на которое барон Мюнхгаузен может приблизиться к центру координат, перемещаясь только на ядрах, с абсолютной или относительной погрешностью не более 10^{-6} .

Примеры

стандартный ввод	стандартный вывод
4 1 0 0 1 -1 0 0 -1	0.0000000000
5 0 3 1 2 1 3 1 4 2 3	0.7071067812

Замечание

Иллюстрация ко второму примеру:



Задача I. Квадрокоптер

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В компании NAUMEN проводится тестирование новейшего ПО для квадрокоптеров. Вове, работающему в NAUMEN, было поручено запустить квадрокоптер на клетчатом поле. Но вот беда — подул сильный ветер, и квадрокоптер стал неуправляемым! Вове нужно как можно быстрее его поймать.

Клетчатое поле имеет h рядов, пронумерованных от 1 до h , и w столбцов, пронумерованных от 1 до w . Каждая клетка имеет *болотистость* (целое число от 1 до 999 999) и *направление ветра* (одно из четырёх: вверх, вниз, влево или вправо).

Вова начинает в клетке в ряду r_1 и столбце c_1 и ходит по полю как хочет. Он может переместиться на клетку, имеющую общую сторону с его текущей клеткой, на это у него уходит количество секунд, равное болотистости клетки, куда он хочет переместиться. Более формально, чтобы переместиться на соседнюю клетку с болотистостью x , Вова ждёт x секунд, после чего моментально переходит на выбранную клетку. Выходить за пределы поля Вова не может. Вова также может решить никуда не ходить, а стоять на текущей клетке в течение произвольного количества времени.

Квадрокоптер начинает в клетке в ряду r_2 и столбце c_2 и перемещается по направлению ветра со скоростью 1 клетка в секунду. Более формально, по прошествии каждой секунды квадрокоптер моментально перелетает на соседнюю клетку в направлении ветра. Направление вверх соответствует уменьшению номера ряда, вниз — увеличению номера ряда, влево — уменьшению номера столбца, вправо — увеличению номера столбца. Если ветер в текущей клетке направлен за пределы поля, то квадрокоптер улетает с поля, после чего его нельзя поймать.

Как только Вова окажется в одной клетке с квадрокоптером, Вова сможет поймать его. При этом если Вова оказывается в клетке в тот же момент, когда квадрокоптер улетает с этой клетки, Вове не удастся его поймать. Определите, сможет ли Вова поймать квадрокоптер, и если да, то найдите наименьшее возможное количество секунд, за которое он сможет это сделать.

Формат входных данных

В первой строке через пробел заданы два целых числа h и w — высота и ширина поля ($1 \leq h, w \leq 250; w \cdot h \neq 1$).

Во второй строке через пробел заданы два целых числа r_1 и c_1 — начальное положение Вовы ($1 \leq r_1 \leq h; 1 \leq c_1 \leq w$).

В третьей строке через пробел заданы два целых числа r_2 и c_2 — начальное положение квадрокоптера ($1 \leq r_2 \leq h; 1 \leq c_2 \leq w$). Гарантируется, что Вова и квадрокоптер начинают в разных клетках.

Далее идут h строк, задающих болотистость клеток. В i -й из них через пробел заданы w чисел от 1 до 999 999 включительно; j -е из чисел равно болотистости клетки в ряду i в столбце j .

Далее идут h строк, задающих направление ветра в клетках. В i -й из них даны w заглавных букв «U», «D», «L» или «R», не разделённых пробелом; j -й из символов соответствует направлению ветра в ряду i в столбце j . Символ «U» означает направление вверх, «D» — вниз, «L» — влево, «R» — вправо.

Формат выходных данных

Если Вова не сможет поймать квадрокоптер, выведите единственное число -1 . Иначе, выведите наименьшее возможное количество секунд, за которое Вова сможет поймать квадрокоптер.

Примеры

стандартный ввод	стандартный вывод
2 7 1 1 1 4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 RDRDRDR URURURU	7
2 6 1 1 1 4 1 1 1 1 1 1 1 1 1 1 1 1 RDRDRD URURUR	-1
3 3 2 2 1 1 8 9 8 8 8 8 8 8 8 RRD URD ULL	9

Задача J. Интересные разговоры

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Алиса и Боб любят принимать у себя гостей, а особенно любят слушать, о чем говорят их гости между собой. Сегодня у Алисы праздник — она в очередной игре победила Боба при том, что он играл оптимально. Это ли не повод отпраздновать?

На праздник Алиса позвала n друзей, которых рассадила за круглым столом и пронумеровала их по часовой стрелке. Для каждого друга Алиса определила его *интересность* целым положительным числом a_i . Как только все расселись за столом, друзья начали общаться между собой, при том только с теми, кто сидит непосредственно слева или справа от них. Чтобы понять, какие разговоры стоит слушать, а какие нет, Алиса определила *интересность* i -го разговора как $|a_i - a_{i+1}|$ при $1 \leq i < n$ и как $|a_n - a_1|$ при $i = n$. Чем это значение больше, тем разговор интереснее.

Теперь, чтобы снова не расстроить Боба и при этом получить удовольствие от подслушивания, Алиса хочет разделить все разговоры на два непустых множества так, чтобы каждый разговор попал только в одно множество и суммарные интересности разговоров в каждом множестве были равны. Помогите Алисе найти эти множества.

Формат входных данных

Первая строка содержит одно целое число n ($3 \leq n \leq 100000$).

Вторая строка содержит n целых чисел a_1, a_2, \dots, a_n — интересности друзей ($1 \leq a_i \leq 10^9$).

Формат выходных данных

В первой строке выведите «YES», если разговоры можно разбить на два множества так, как хочет Алиса, или «NO» в противном случае. В случае «NO» дальше выводить ничего не нужно.

Во второй строке выведите m — размер первого множества.

В третьей строке выведите через пробел p_1, p_2, \dots, p_m — номера разговоров в первом множестве.

В четвертой строке выведите k — размер второго множества.

В пятой строке выведите через пробел q_1, q_2, \dots, q_k — номера разговоров во втором множестве.

Пример

стандартный ввод	стандартный вывод
3	YES
1 2 3	2
	1 2
	1
	3

Задача К. Вася и прогулка по бесконечной дороге

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вася любит гулять по необычным местам. Однажды он набрёл на красивую бесконечную дорогу, по которой он, конечно же, решил прогуляться. Вася передвигался по дороге с постоянной скоростью, и каждый час проходил мимо нескольких столбов. Примечательно, что ровно в конце каждого часа Вася проходил мимо столба. Будем называть столб, мимо которого проходил Вася в конце i -го часа, i -м *интересным* столбом.

Столбы расположены равномерно вдоль дороги, и у каждого столба есть номер. Номера столбов — подряд идущие целые числа. Вася, не сбавляя скорости, записывал себе в блокнот все номера интересных столбов.

Сильный ветер сорвал лист блокнота Васи, на котором был записан номер k -го интересного столба. Вернувшись домой с прогулки, Вася захотел снова записать его в блокнот, но он не может вспомнить, с какой скоростью он шёл по прекрасной бесконечной дороге, чтобы восстановить потерянный номер. Помогите Васе привести его записи в порядок.

Формат входных данных

В единственной строке задано число k ($0 \leq k \leq 10^5$).

Формат выходных данных

Выведите номер k -го интересного столба.

Примеры

стандартный ввод	стандартный вывод
2019	20191027
2020	20201101

Задача L. Прыжок с парашютом

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Кирилл выиграл сертификат на прыжок с парашютом и решил не медлить. Он приехал в прямоугольную декартовую систему координат, в которой земля и воздух разделены прямой $y = 0$ м, воздух находится сверху (в полуплоскости $y \geq 0$ м), земля — снизу. Гравитация направлена вниз (в направлении уменьшения координаты y).

Кирилл поднялся в точку (x, y) и начал падать. Кирилл падает с ускорением свободного падения ровно 10 м/с^2 (да, в этом странном мире парашют не замедляет падение, а только смягчает приземление). Изначально вертикальная скорость Кирилла равна 0 м/с . А вот горизонтальной скоростью он может свободно управлять: в любой момент времени Кирилл может поменять свою горизонтальную скорость на любую от -1 м/с до 1 м/с .

Кирилл хочет пробыть в воздухе как можно дольше. К счастью, он заметил в воздухе n облаков. Кирилл точно определил, что i -е облако находится в точке (x_i, y_i) и имеет плотность c_i . Если Кирилл попадёт в точку, где находится i -е облако, то полностью остановится и следующие c_i секунд провисит неподвижно в этой точке. После чего он начнёт снова падать с начальной вертикальной скоростью 0 м/с и сможет менять горизонтальную скорость на любую от -1 м/с до 1 м/с .

Определите наибольшее возможное время, которое Кирилл может падать. Его падение заканчивается в тот момент, когда он касается прямой $y = 0$ м.

Формат входных данных

В первой строке задано единственное целое число n — количество облаков ($0 \leq n \leq 50000$).

Во второй строке через пробел задано два целых числа x и y — начальные координаты Кирилла в метрах ($-10000 \leq x \leq 10000$; $1 \leq y \leq 10000$).

Далее идут n строк, в i -й из которых через пробел заданы три целых числа x_i , y_i и c_i — координаты очередного облака в метрах и его плотность ($-10000 \leq x \leq 10000$; $0 < y_i < y$; $1 \leq c_i \leq 10000$). Гарантируется, что координаты всех облаков различны.

Формат выходных данных

Выведите единственное число — наибольшее возможное время падения в секундах с абсолютной или относительной погрешностью не более 10^{-4} .

Примеры

стандартный ввод	стандартный вывод
5 0 80 -1 30 2 -1 60 1 0 40 10 1 20 1 1 50 1	20.863703305
0 100 100	4.472135955