Задача A. Алгоритм Куна нам только снится

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

«Два года мы готовились, а вы тренировались, и вот традиционные чемпионаты возвращаются!» — с такой фразы начал очередную свою лекцию Руслан. Его лицо выражало некоторую усталость.

«Мы надеемся, что у участников стало больше опыта в решении задач!» — Последние несколько ночей он составлял задачи для чемпионата. Он хотел, чтобы они решили задачи, решили их все!

Вместе с тем на лекции присутствовали два друга, Кузя и Шмузя. Кузе очень нравилось заниматься олимпиадным программированием. Особенно ему нравились лекции Руслана.

Шмузя тоже любил программирование, но при этом был тем еще оболтусом: всю ночь он обсуждал алгоритм сортировки пузырьком в комментариях на некотором сайте e-minn.ru и изрядно при этом устал.

Неудивительно, что когда Руслан вошел в аудиторию и громогласно заявил о возвращении легендарных чемпионатов ПетрГУ, Шмузя разве что не похрапывал, а на его лице уже успел отпечататься конспект прошлой лекции.

Сквозь сон Шмузя слышал отрывки слов Руслана:
> «...ритм Куна, который позволяет решать задачу о максимальном паросочетан...»
> «...пишем определение: двудольный граф — это граф, вершины которого можно разбить на два множества...»
> «...при этом каждое ребро графа соединяет какую-то вершину из первого множества с какой-то вершиной второ... »
> «...ажно! Не существует ребер, соединяющих вершины из одного множества... »

Несмотря на Кузины тычки в бок Шмузя продолжал сладко спать. Лекция его убаюкивала:
> «...паросочетанием называется такой набор рёбер графа, что никакие два из них не имеют общей вершины...»
> «...айте назовем мощностью паросочетания количество рёбер в нём...»
> «...наибольшим паросочетанием назовём паросочетание, мощность которого максимальна среди всех возможных в данном графе...»

В один момент Шмузя всхрапнул особенно громко. Руслан наконец-то обратил внимание на своего спящего падавана. «Шмузя, – раздался громогласный клич, – может быть ты нам скажешь, какое паросочетание является наибольшим в данном графе?».

Шмузя резко поднял голову, растерянно протер глаза и окинул аудиторию взглядом. В глазах лектора сверкала недобрая искра, а Кузя тихонько сползал под парту, как бы пытаясь спрятаться.

На большой доске позади Руслана красовался нарисованный мелом неориентированный граф. Шмузя заметил, что все вершины графа распределены на два множества, в каждом из которых ровно n вершин.

«Ребро в графе между i-ой вершиной первого множества и j-ой вершиной второго множества существует только тогда, когда наибольший общий делитель чисел i и j равен 1» – донесся до Шмузи быстрый шепот Кузи из-под парты.

У Шмузи есть только секунда, чтобы решить эту задачу! Помогите ему!

Формат входных данных

В единственной строке содержится число n ($1 \leqslant n \leqslant 3 \cdot 10^5$).

Формат выходных данных

В первой строке выведите число k — количество ребер в максимальном паросочетании.

В следующих k должно содержаться описание ребер паросочетания: в i-ой строке через пробел выведите два числа a_i и b_i — номера вершин начала и конца i-го ребра.

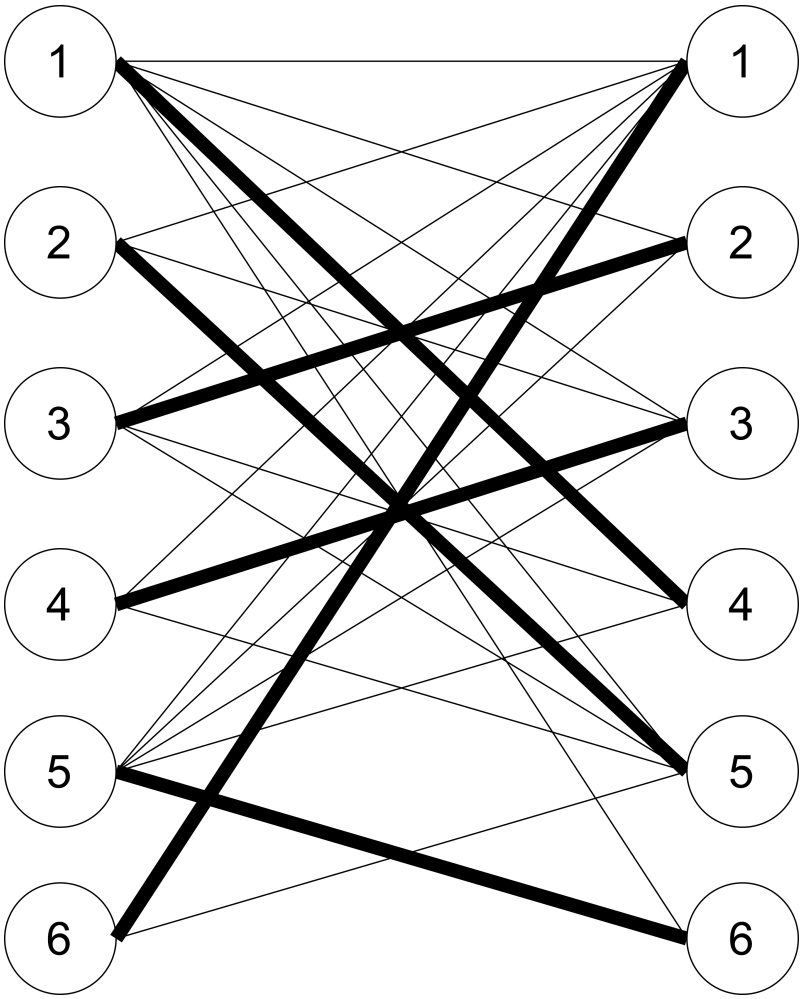
Если ответов несколько, выведите любой из них.

Примеры

стандартный ввод	стандартный вывод
1	1 1 1
2	2 1 2 2 1
6	6 1 4 2 5 3 2 4 3 5 6 6 1

Замечание

На рисунке ниже изображен граф для $n = 6$, а жирным выделены ребра, входящие в ответ. Условия задачи соблюдаются, так как все ребра соединяют только взаимнопростые числа, выбранные ребра не пересекаются по вершинам, и не существует способа выбрать большее количество рёбер. Обратите внимание, что есть и другие варианты ответа, но все они будут содержать такое же число выбранных рёбер.



Задача B. Пустите нас в 211

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

Перед чемпионатом все усердно готовятся. А те, кто обучаются в КТП, тренируются в 211 кабинете айтипарка. К сожалению или к счастью, не всем участникам КТП дают ключ, а некоторые обучающиеся могут находится в кабинете только в присутствии преподавателя. И вот как раз за день до олимпиады, когда все заняты, ребята, которым запрещено находится в 211 без преподавателя, решили потренироваться и написать контест.

Чтобы поменьше напрягать преподавателей, они объединились и решили написать тренировку в промежуток времени с L по R, без перерывов.

Про каждого преподавателя они заранее узнали, в какой промежуток времени он может присутствовать в 211 и какой дискомфорт он испытает в связи с этим.

Ребята хотят напрягать преподавателей как можно меньше. Так как эмоции преподавателей никак не взаимосвязаны, то общий дискомфорт преподавателей не суммируется и равен наибольшему дискомфорту, который испытал кто-то из приглашенных преподавателей. Помогите ребятам выбрать преподавателей, которых они попросят прийти в 211, чтобы на протяжении всей тренировки к кабинете был кто-то из преподавателей, а максимальный дискомфорт был как можно меньше.

Формат входных данных

В первой строке дано число $1 \leqslant n \leqslant 10^5$ — количество преподавателей в КТП. Далее в n строчках дано описание каждого преподавателя, состоящее из 3 чисел $0 \leqslant l_i, r_i, w_i \leqslant 10^{18}$ — временной промежуток, на который может прийти i-ый преподаватель, и его дискомфорт w_i в связи с приходом. В последней строке дано два числа $0 \leqslant L < R \leqslant 10^{18}$ — промежуток времени, на которых хотят прийти ребята.

Формат выходных данных

Если ни при каких обстоятельствах, ребята не смогут пробыть в 211 промежуток времени с L по R, то выведите “-1”. Иначе, в первой строке выведите кол-во преподавателей, которых попросят ребята. Во второй строке выведите индексы этих преподавателей, в любом порядке. Если ответов несколько, разрешается вывести любой.

Пример

стандартный ввод	стандартный вывод
9 11 12 3 0 8 9 6 9 1 1 12 0 0 7 11 5 8 2 9 11 10 5 11 9 5 8 5 0 10	2 2 4

Задача C. По просьбам трудящихся

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

«Хотим хлеба и геометрию!» — кричали они.

На протяжении всей подготовки чемпионата ПетрГУ многие потенциальные участники шутили про уровень сложности задач, про их возможную тематику. Естественно, авторы подошли к этому очень скептически, но решили удовлетворить некоторые забавные просьбы.

Как-то на лекции по геометрии в выездном лагере по программированию, который обычно проводится в поселке Э., некий ученик Н. заскучал, и нарисовал много точек в своей тетради в клеточку, причем каждая точка находилась ровно на пересечении тетрадных линий. Как только сосед его одернул, Н. осознал сделанное и задумался, как нарисовать прямую, чтобы количество точек, для которых есть симметричная точка, относительно этой прямой среди нарисованных, было как можно больше. Помогие Н. решить его проблему, чтобы он слушал лекцию дальше.

Формат входных данных

В первой строке дано количество точек ($1 \leqslant n \leqslant 10^3$). В следующих n строках даны координаты x и y точек ($-10^9 \leqslant x, y \leqslant 10^9$).

Формат выходных данных

Выведите максимальное количество точек симметричных относительно одной прямой.

Пример

стандартный ввод	стандартный вывод
8 2 1 2 3 4 2 5 2 4 -1 6 5 8 1 8 3	6

Замечание

В примере прямая, для которой есть 6 симметричных точек, имеет уравнение $y = 2$. Обратите внимание, что точка на прямой сама себе симметрична относительно этой прямой.

На рисунке ниже показаны несколько вариантов прямой с указанием количества симметричных точек, ответ на задачу изображен на рисунке 1.

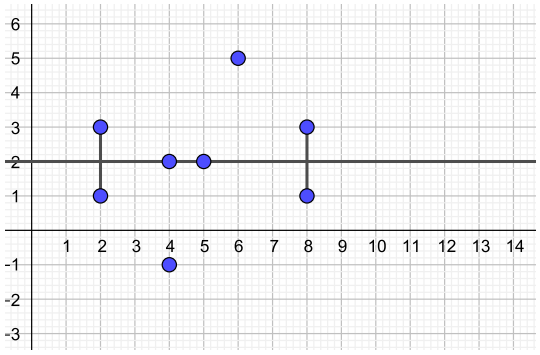


Рис. 1: прямая с 6 симметричными точками

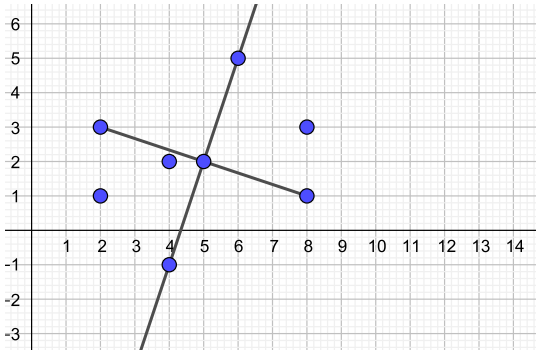


Рис. 2: прямая с 5 симметричными точками

Задача D. Решишь квадратное уравнение?

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

В КТП ежегодно происходит набор новых учеников. Как правило, туда приходят люди с 8 класса и старше, так как на занятиях надо как минимум знать, как решать квадратное уравнение, ведь это одна из самых важных основ математики.

Но даже если вы не умеете решать квадратное уравнение, мы вас научим. Квадратное уравнение — это приравненный к нулю многочлен второй степени. Стандартная форма обычно записывается следующим образом: $ax^2 + bx + c = 0$, где $a \neq 0$. Чтобы его решить, чаще всего пользуются таким понятием как дискриминант.

«В общем случае дискриминант многочлена $p(x) = a_0 + a_1x + \cdots + a_nx^n, a_n \neq 0$, есть произведение

$$D(p) = a_n^{2n-2} \prod_{i<j} (\alpha_i - \alpha_j)^2$$

где $\alpha_1, \alpha_2, \ldots, \alpha_n$ — все корни многочлена (с учётом кратностей) в некотором расширении основного поля, в котором они существуют.»[Википедия].

В частном случае, когда степень многочлена равна двум, его можно посчитать по формуле $D = b^2 - 4ac$.

В чем же важность дискриминанта? Если он равен нулю, то можно утверждать, что корни совпадают, а если он меньше нуля, то действительных корней нет вовсе. Если же дискриминант положителен, то существует два различных действительных корня, которые можно вычислить по формулам: $x_{1,2} = \frac{-b \pm \sqrt{D}}{2a}$.

На зачете за первый семестр первому году обучения предложили сдать в контесте задачу по решению квадратного уравнения. Преподаватели заботливо сгенерировали хорошие тесты, каждый из которых представлял из себя три ненулевых целых числа a, b, c, задающие квадратное уравнение с целыми корнями. Участникам зачета осталось только написать простые формулы, записанные на лекциях со школы. Но в тестирующей системе произошел сбой, вместо этих трех чисел теперь на вход подается сразу корень квадратный из дискриминанта для этого уравнения, при этом он оказался всегда целый. Тем не менее, сдать задачу надо, то есть надо вывести два различных целых решения квадратного уравнения, которое могло иметь такой дискриминант.

Формат входных данных

На вводе дано одно целое число $1 \leqslant (d = \sqrt{D}) \leqslant 10^{16}$.

Формат выходных данных

Выведите целые корни любого квадратного уравнения, у которого ненулевые коэффициенты, а корень из дискриминанта есть d. При этом корни должны влезать в 64-битный знаковый тип данных.

Пример

стандартный ввод	стандартный вывод
1080	89 17

Замечание

В первом примере могло быть загадано уравнение $15x^2 - 1590x + 22695 = 0$. Дискриминант равен $D = 1590^2 - 4 \cdot 15 \cdot 22695 = 1080^2$. А корни соответственно равны

$$x_{1,2} = \frac{1590 \pm 1080}{30} = 89, 17.$$

Задача E. Плейлист для двоих

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

Работать в компании $\leqslant$ работать одному

После тяжелого дня, в очередной раз сидя вдвоем в кофейне, Оля и Свят — двое участников КТП — решили поработать. У каждого свои дела, но в компании родственной души работать приятнее. Чтобы быть на одной волне, они решили слушать один и тот же плейлист, но каждый в своих наушниках.

Теперь ребятам нужно выбрать музыку, которая будет их синхронизировать и вдохновлять на подвиги.

Так как мы все же ~~занудЫ~~ программисты, давайте разберемся что такое «плейлист».

Плейлист — набор композиций, в который вошли некоторые из списка доступных на устройстве. Иначе говоря это какое-то подмножество множества всех доступных на устройстве композиций.

При этом для каждой композиции известны некоторые численные характеристики a и b, равные удовлетворению, какое эта композиция принесет Оле и Святу соответственно. К несчастью, удовлетворение может быть как положительным, так и отрицательным.

Выгодой плейлиста для конкретного участника КТП будем называть сумму удовлетворений от прослушивания каждой композиции этого плейлиста.

Назовем плейлист *удачным*, если его выгода для Оли отличается от выгоды для Свята не более чем на T. Иначе это слишком несбалансированный плейлист и кто-то из ребят будет грустить.

Чтобы никому из слушателей не было обидно и комфорт обоих участников небольшого кофейного коворкинга оказался примерно одинаков, было принято решение среди всех удачных плейлистов найти такой, что сумма выгод от этого плейлиста для первого и второго участника КТП будет максимальной.

Помогите Оле и Святу составить такой плейлист из имеющегося набора композиций.

Формат входных данных

В первой строке дано количество композиций, доступных на устройстве $1 \leqslant n \leqslant 10^3$. Далее в n строках идет описание композиций. В i-ой строке содержится два целых числа a_i b_i разделенных пробелом ($-100 \leqslant a_i, b_i \leqslant 100$) — удовлетворение от прослушивания i-ой композиции для Оли и Свята соответственно. В последней строке записано ограничение на разность выгоды $0 \leqslant T \leqslant 100$.

Формат выходных данных

В единственной строке выведите одно целое число — наибольший суммарный комфорт, который могут получить ребята собрав сбалансированный плейлист. Если ребятам невыгодно в принципе слушать музыку, выведите ноль.

Примеры

стандартный ввод	стандартный вывод
6 -8 9 9 -8 1 2 2 1 -5 100 5 5 2	18
2 -100 -10 5 -10 50	0
2 -5 10 -100 100 15	5

Задача F. А у кого-то есть время читать книжки....

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

Все олимпиадники — очень занятые люди. Иногда настолько, что им даже некогда почитать книжку для души. То у них тренировки, то лекции, то чемпионаты. А если они и читают, то это техническая книжка по программированию или математике.

Чтобы исправить эту оплошность мы предлагаем вам почитать... прямо сейчас! Раз это олимпиада, то книжка будет написана математиком, но что бы почитать для души, это будет художественная литература.

Представляем вам отрывок из произведения «Алиса в стране чудес», написанное Льюисом Кэрроллом:

«There was a table set out under a tree in front of the house, and the March Hare and the Hatter were having tea at it: a Dormouse was sitting between them, fast asleep, and the other two were using it as a cushion, resting their elbows on it, and talking over its head. 'Very uncomfortable for the Dormouse,' thought Alice; 'only, as it's asleep, I suppose it doesn't mind.'

The table was a large one, but the three were all crowded together at one corner of it: 'No room! No room!' they cried out when they saw Alice coming. 'There's plenty of room!' said Alice indignantly, and she sat down in a large arm-chair at one end of the table.

'Have some wine,' the March Hare said in an encouraging tone.

Alice looked all round the table, but there was nothing on it but tea. 'I don't see any wine,' she remarked.

'There isn't any,' said the March Hare.

'Then it wasn't very civil of you to offer it,' said Alice angrily.

'It wasn't very civil of you to sit down without being invited,' said the March Hare.

'I didn't know it was your table,' said Alice; 'it's laid for a great many more than three.'

'Your hair wants cutting,' said the Hatter. He had been looking at Alice for some time with great curiosity, and this was his first speech.

'You should learn not to make personal remarks,' Alice said with some severity; 'it's very rude.'

The Hatter opened his eyes very wide on hearing this; but all he said was, 'Why is a raven like a writing-desk?'

'Come, we shall have some fun now!' thought Alice. 'I'm glad they've begun asking riddles. — I believe I can guess that,' she added aloud.

'Do you mean that you think you can find out the answer to it?' said the March Hare.

'Exactly so,' said Alice.

'Then you should say what you mean,' the March Hare went on.

'I do,' Alice hastily replied; 'at least — at least I mean what I say — that's the same thing, you know.'

'Not the same thing a bit!' said the Hatter. 'You might just as well say that "I see what I eat"is the same thing as "I eat what I see"!'

'You might just as well say,' added the March Hare, 'that "I like what I get"is the same thing as "I get what I like"!'

'You might just as well say,' added the Dormouse, who seemed to be talking in his sleep, 'that "I breathe when I sleep"is the same thing as "I sleep when I breathe"!'

'It is the same thing with you,' said the Hatter, and here the conversation dropped, and the party sat silent for a minute, while Alice thought over all she could remember about ravens and writing-desks, which wasn't much. »

Как истинным айтишникам, нам интересно, в скольких словах в этом отрывке встретилась подстрока ''IT'' без учета регистра.

Формат входных данных

В единственном тесте будет текст, данный в условии.

Формат выходных данных

Выведите в единственной строке число, являющееся количеством слов в тексте в которых присутствует подстрока “IT”. Заметим, что эта подстрока не чувствительна к регистру т.е. “IT”, “It” или “iT” тоже являются верными подстроками.

Задача G. Очепятки не порок

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

Некоторые участники КТП часто очепятываются, когда пишут текст, но это не так страшно, особенно, елси текст все еще можно расшифровать. Чаще всего это можно сделать, елси все слова содержат те же бкувы, какие и должны были содержать, но возможно, в другмо порядке. Если же слова содержат иные бкувы, или некоторые из них остутствуют, то расшифровать сложнее. Будем считать, что слово написано условно правильно, елси оно содержит все те бкувы, что и должно содержать, но возможно в другом порядке. Если же в слове набор букв неверный, то слово считается написанным с ошибкой.

В очередной раз перечитывая условие задачи, авторы обнаружили массу опечаток, которые они ринулись срочно исправлять. Но, прежде чем это делать, они вспомнили несколько фактов про восприятие условий. Во-первых, первый абзац как правило никто не читает. Во-вторых, перепутанный порядок букв в слове чаще все воспринимают нормально, а вот слова с ошибками просто игнорируют. В-третьих, если в любой части условия проигнорировать более k слов подряд, то задача становится максимально непонятной и это критично плохо, поэтому некоторые слова с ошибкой надо исправить. Но так как исправлять условие надо быстро, то надо исправить наименьшее количество слов с ошибками, чтобы в тексте не встретилось более k неверных слов подряд.

Формат входных данных

В первой строке выведены 3 числа — n, m, k ($1 \leqslant n, m \leqslant 10^4$, $1 \leqslant k \leqslant 100$) - количество слов в словаре, количество слов в тексте и максимальное количество слов с ошибкой, которым позволительно появляться в тексте. Далее следуют n строк, в каждой строке находится s_i слово из словаря (длина слова не превосходит 30 символов). Считается, что других слов, кроме слов из словаря, в языке, на котором написано условие, нету. В следующей строке находится текст условия, представляющий собой m слов через пробел (длина слова не превосходит 30 символов). Все слова содержат только строчные латинские буквы.

Формат выходных данных

Выведите одно число - минимальное количество слов которое нужно исправить, чтобы текст содержал не более k подряд идущих слов с ошибкой.

Пример

стандартный ввод	стандартный вывод
10 10 2 acm i love tasks solve coffee drinking during some contest i love sovle some acn taksk during drinkin som cofee	1

Замечание

Опечатки в условии: бкувы = буквы, очепятываются = опечатываются, елси = если, ругмо = другом, остутствуют = отсутствуют

Задача H. Бильярд в графе

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

Бильярдные траектории — любимая тема некоторых участников ~~Жюри~~ КТП. Формальное определение сложно, поэтому здесь мы ограничимся некоторым упрощением. "Основная" идея в том, что есть некоторый процесс, который однозначно задается текущим состоянием и тем, как мы в него пришли. Приведем несколько примеров. Чтобы определить движение бильярдного шара, нужно знать его текущую точку местоположения и вектор скорости. Рейтинг олимпиадника в КТП после соревнования зависит от его положения в текущем контесте и рейтинга до чемпионата. Настроение олимпиадника на чемпионате зависит от качества задач и от того, с каким настроением он пришел на чемпионат.

В этот раз было решено определить бильярд на полном графе. Некая точка перемещается по вершинам графа, при этом, то, куда она пойдет зависит от вершины, в которой она сейчас находится, и ребра, по которому она пришла. Так как ребро задаёт предыдущую вершину, то будем говорить, что номер следующей посещенной вершины зависит от номера текущей и предыдущей вершин. Чтобы не думать, как устроены эти переходы, жюри уже посчитало для вас матрицу перехода ,где A_{ij} означает что если мы пришли в вершину j из вершины i, то следующей вершиной, в которую мы пойдем, будет A_{ij}.

Дано q запросов вида x y k — если мы стартуем из x в y, и совершаем k операций перехода. Итерация из x в y считается нулевой итерацией.

На каждый такой запрос вам нужно ответить, в вершине с каким номером мы окажемся через k итераций?

Формат входных данных

В первой строке даны два целых числа n и q ($1 \leqslant n \leqslant 500$, $1 \leqslant q \leqslant 10^5$). Далее идет n строк, в каждой из которых содержится n целых чисел разделенных пробелами: описание матрицы перехода A размера $n \times n$. На пересечении i-й строки и j-го столбца матрицы A находится элемент матрицы перехода A_{ij} ($1 \leqslant a_{ij} \leqslant n$) Далее идет q строк, в каждой из которых содержится три целых числа x, y, k ($1 \leqslant x, y \leqslant n, 10^9 \leqslant k \leqslant 10^{18}$) — вершина из которой мы стартовали, в какую вершину мы из нее пойдем и количество операций перехода.

Формат выходных данных

Выведите q строк, в i-й строке строке выведите единственное число m_i ($1 \leqslant m \leqslant n$) — ответ на i-й запрос.

Примеры

стандартный ввод	стандартный вывод
2 2 2 2 2 1 2 1 1000000005 2 1 1000000004	1 2
9 8 7 4 3 8 4 3 3 6 8 9 4 4 6 2 8 4 7 8 7 1 4 7 7 7 9 2 4 1 7 7 8 7 9 8 4 6 8 8 7 5 7 3 4 5 1 6 4 8 3 4 4 7 6 1 1 1 5 5 5 7 4 5 9 3 3 2 1 2 2 3 9 9 6 4 7 6 2 6 7 9 7 9 6 1000000004 8 6 1000000004 9 4 1000000004 6 7 1000000002 9 3 1000000001 5 1 1000000004 5 3 1000000003 2 8 1000000003	7 7 5 5 7 4 5 7

Замечание

Рассмотрим второй пример. Стартуя из 9 в 6 мы получим такую последовательность: 9 6 6 4 3 7 9 9 7 7 4 5 7 4 5 7 4 5 7 4 5 7 4 5 ... Легко заметить, что появляется период 7 4 5, что по сути делает последовательность периодической с предпериодом 9 6 6 4 3 7 9 9 7.

Задача I. Ещё больше кофе

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

Одна особенная команда КТП очень хочет пройти на финал ACM ICPC. Так сильно хочет, что пишет тренировки каждый день. Но пишет она их плохо, по этому каждый раз приходится писать в новом месте.

Также участники команды регулярно не высыпаются перед тренировкой. Как истинные программисты, они решают эту проблему при помощи кофе!

Команде нужно МНОГО кофе, чтобы написать тренировку хорошо. чтобы раздобыть заветный напиток, они отправляют самого главного кофемана – Тимошу – пройтись по ближайшим кофейням и купить столько кофе, сколько тот сможет найти.

Вместе с тем, город П., в котором живут наши герои, представляет собой набор перекрестков, расположенных в узлах прямоугольной сетки. Все в городе очень любят кофе, так что кофейня есть буквально на каждом перекрестке.

Тимоша знает, что место написания тренировки находится находится в точке с координатами (x, y).

При этом, в связи «прямоугольностью» города, расстояние от места написания тренировки до кофейни с координатами (c_x, c_y) определяется как $|x - c_x| + |y - c_y|$. Как известно маленький город П. является практически вторым Нью-Йорком, так что будем называть такое расстояние *манхэттенским*.

Тимоше нужно купить как можно больше кофе, чтобы команда уж точно написала тренировку хорошо. При этом он не хочет уходить далеко от места написания тренировки, так как боится потеряться. Выражаясь формально, Тимоша не будет посещать кофейни, расстояние до которых от текущего места проведения тренировки превышает величину h.

Тимоша заядлый кофеман, так что в каждой кофейне его знают в лицо. При этом у него не самая лучшая репутация, так что в каждой кофейне есть лимит на кол-во чашек кофе, которое ему продадут. Формально, в кофейне с координатами c_x, c_y можно купить не более a_{xy} чашек кофе в день.

Каждый день Тимоша посещает все кофейни, которые располагаются неподалеку от места проведения тренировки. При этом иногда лимит на продажу кофе в какой-то конкретной кофейне меняется. Ваша задача понять, какое наибольшее кол-во чашек кофе может купить команда в конкретный день.

Формально, вам будет предложено ответить на q запросов, каждый из которых — либо вылазка за кофе или смена лимита чашек в одной из кофеен. Дано двумерное поле n на m, заполненное числами, и два вида запросов на нем:

- 0 тип. Параметры x y val — лимит в кофейне с координатами x и y изменяется на новое значение val;
- 1 тип. Параметры x y h — координаты места проведения контеста и наибольшее расстояние на которое готов удаляться Тимоша. Необходимо посчитать общее число чашек кофе, которое можно купить в кофейнях, до которых манхэтенское расстояние от места проведения не превосходит h. При этом гарантируется что
  a) $h \leqslant x, y$;
  b) $x + h \leqslant n$;
  c) $y + h \leqslant m$.

Формат входных данных

В первой строке находятся 3 числа n, m, q ($1 \leqslant n, m \leqslant 1000$, $1 \leqslant q \leqslant 3 \cdot 10^5$). Далее в n строках перечисляются m чисел a_{ij} — исходные лимиты в каждой кофейне, расположенной в координатах (i, j). Далее идут q строк, в каждой 4 числа: тип запроса (0 или 1) и параметры этого запроса.

Формат выходных данных

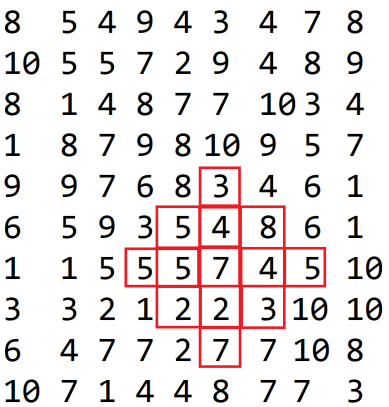
Для каждого запроса 1 выведите одно число — максимальное кол-во чашек, которое может купить Тимоша.

Пример

стандартный ввод	стандартный вывод
10 9 4 8 5 4 9 4 3 4 7 8 10 5 5 7 2 9 4 8 9 8 1 4 8 7 7 10 3 4 1 8 7 9 8 10 9 5 7 9 9 7 6 8 3 4 6 1 6 5 9 3 5 4 8 6 1 1 1 5 5 5 7 4 5 10 3 3 2 1 2 2 3 10 10 6 4 7 7 2 7 7 10 8 10 7 1 4 4 8 7 7 3 1 7 6 2 1 8 6 2 1 5 5 4 0 4 7 5	60 62 240

Замечание

Пояснение к примеру, к первому запросу.
Стартуя из ячейки с координатами $(7, 6)$ мы можем обойти выделенную область, скупив 60 чашек кофе.



Задача J. Логотип КТП

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

Совсем недавно Руслану в голову забрела мысль, что надо бы обновить старый пикселизированный логотип КТП и сделать много классной сувенирной продукции. Помимо сложных художественных решений, необходимых для создания запоминающегося образа нового логотипа, было необходимо также решить массу технических проблем. Прежде всего, логотип должен быть перерисован в векторном формате. Так как далеко не все умеют легко работать в векторных графических редакторах, работу передали профессиональному графическому дизайнеру.

Одной из главный творческих идей заказчика являлось оформление низа логотипа каёмкой из усечённых треугольников, одна сторона которых лежит на оси абсцисс.

Дизайнер нарисовал треугольники и хочет их усечь, но он не уверен, как именно. Началось традиционное общение с заказчиком и ежедневное обновление ТЗ. Каждый день Руслан присылал дизайнеру, по какой y-координате надо обрезать треугольники, а дизайнер в ответ присылал ему результат. По неведомым никому причинам, Руслану крайне важна площадь обрезанной части.

Формально, даны n непересекающихся треугольников, основания которых лежат на оси абсцисс, и q запросов вида: узнать суммарную площадь треугольников, отсекаемых прямой y $= a_i$.

Формат входных данных

В первой строке ввода дано два целых числа n и q ($1 \leqslant n, q \leqslant 10^5$).

Далее идут n строк, в каждой из которых содержится по 6 целых чисел $x_1, y_1, x_2, y_2, x_3, y_3$ ($0 \leqslant x_1, x_2, y_1 x_3 \leqslant 10^9, 1 \leqslant y_1, x1 \leqslant x2 \leqslant x3, y_1 = y_3 = 0$) – координаты каждого треугольника. Гарантируется, что треугольники всегда пересекаются только в крайних точках.

Далее дано q строк, в каждой из которых содержится целое число $a_i (1 \leqslant a_i \leqslant 10^9)$ – прямая для текущего запроса.

Формат выходных данных

На каждую из q строк запроса с параметром прямой выведите соответствующую площадь отсекаемых частей треугольников. Ответ должен быть дан с абсолютной или относительной точностью не менее 10^{-4}.

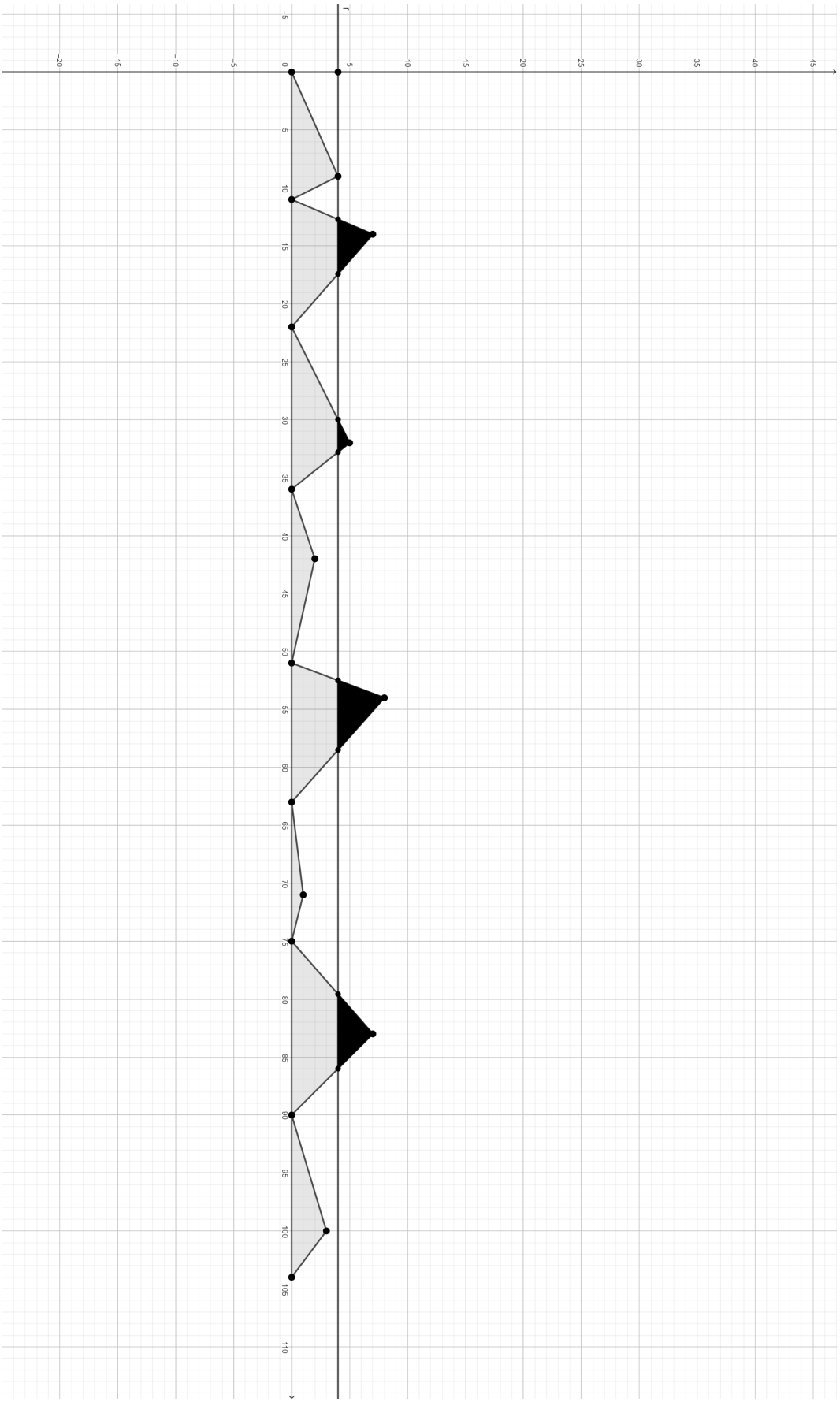
Пример

стандартный ввод	стандартный вывод
8 4 0 0 9 4 11 0 11 0 14 7 22 0 22 0 32 5 36 0 36 0 42 2 51 0 51 0 54 8 63 0 63 0 71 1 75 0 75 0 83 7 90 0 90 0 100 3 104 0 9 8 4 3	-0.000000000000000 -0.000000000000000 30.114285714285714 55.439285714285714

Замечание

Иллюстрация к третьему запросу из первого примера:

(верх там $\longrightarrow$)



Задача K. Фракции КТП, или любителям ТЧ посвящается

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

Фракция выше крови

На текущий момент в КТП есть n участников, у каждого свои интересы, свои хобби, свой круг общения. Как обычно бывает в относительно больших группах, участники часто образуют свои микрогруппы по интересам.

Руслан видит мир через призму олимпиадных задач, поэтому как только он заметил этот факт, он сразу начал строить математическую модель. Конкретно, ему стало интересно, могут ли участники корректно разбиться на группы по общению (фракции) одинакового размера.

После долгих исследований, социальных опросов, построения нейронных сетей и сложного моделирования процессов общения людей на компьютере, Руслан получил следующие данные.

Рассмотрим множество всех интересов всех участников k_i, где каждый интерес задается уникальным простым числом. Теперь каждому участнику КТП можно приписать некоторое число, которое будет характеризовать его интересы: произведение всех простых чисел, соответствующих его интересам.

Дан массив из n чисел – характеристика интересов участников КТП. Обозначим множество всех простых делителей, которые когда-либо встречались, как k_i. Теперь правило, согласно которому фракция может образоваться, выглядит очень просто: у каждого участника фракции должен быть некий интерес (простое число), общий для всех во фракции. Такой интерес можно назвать основным для этой фракции.

На этом этапе Руслан потерял интерес к задаче, поэтому поручил завершение работы вам, в качестве разминки для мозгов.

Зная интересы всех участников КТП, определите, можно ли разбить их на фракции по интересам, чтобы каждый интерес стал основным для своей фракции, а размеры фракций были бы одинаковы.

Формат входных данных

В первой строке дано целое число n ($1 \leqslant n \leqslant 10^5$) – число участников КТП. В следующей строке идут n целых чисел a_i $1 \leqslant a_i \leqslant 10^5$) – описание интересов участника под номером i.

Формат выходных данных

Выведите «YES» (без кавычек), если можно разбить участников КТП на равновеликие фракции по интересам, задействуя все интересы, встречаемые у участников, иначе выведите «NO» (без кавычек).

Примеры

стандартный ввод	стандартный вывод
4 17 34 40 48	YES
3 10 5 8	NO
6 25 10 14 35 70 8	YES

Замечание

В первом примере дано 4 участника со следующим набором интересов: $\{17\}, \{2, 17\}, \{2, 5\}, \{2, 3\}$. Так как имеется всего 4 различных интереса, то каждый из участников представляет собой самостоятельную фракцию.

Во втором примере дано всего 3 участника и 2 интереса (интересы: 2, 5). Трех участников нельзя разбить на две группы одинакового размера.

В третьем примере дано 6 участников с интересами $\{5\}, \{2, 5\}\{2, 7\}, \{5, 7\}, \{2, 5, 7\}, \{2\}$. Соответственно, можно распределить всех участников в три группы по 2 человека следующим образом:

- по интересу «5»: $\{\{5\}, \{2, 5\}\}$;
- по интересу «7»: $\{\{2, 7\}, \{5, 7\}\}$;
- по интересу «2»: $\{\{2\}, \{2, 5, 7\}\}$.

Задача L. Путь домой 2

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

После обряда «Сила Духа» Кузя ездит домой на карельском автомобиле.

В КТП все возвращаются домой поздно, когда всюду пробки. И вот, после тяжелого рабочего дня Кузя заказал такси. В вечернее время город превращается в сеть односторонних дорог, так как в одну из сторон всегда пробка. Кузя не любит, когда не знает, как поедет водитель, но и говорить с ним много он не хочет, поэтому он решил сказать одну фразу “Проедьте через пункт ...”. после которой маршрут водителя бы задался однозначно.

Помогите Кузе определить, может ли он своими словами повлиять на однозначность пути водителя, и если может, то какой пункт он может указать.

Формат входных данных

В первой строке дано 4 числа: n, m, a, b ($1 \leqslant n, m \leqslant 2 \cdot 10^5$,$1 \leqslant a, b \leqslant n$) – количество пунктов в городе, односторонних дорог, номер стартового и конечного пункта путешествия Кузи. В каждой из следующих m строк находится 2 числа s_i, t_i, обозначающих, что можно проехать от пункта s_i в пункт t_i именно в этом направлении. Гарантируется, что между любыми двумя пунктами есть не более одной дороги, соединяющей их напрямую. Гарантируется отсутствие циклов.

Формат выходных данных

В первой строчке выведите «YES», если можно выбрать такой пункт C что элементарный путь от A до B через C однозначен, и «NO» – иначе. При положительном ответе выведите на второй строчке вначале количество выбора пункта C, а далее все такие пункты C через пробел.

Пример

стандартный ввод	стандартный вывод
10 10 1 8 1 2 2 3 3 4 4 5 5 6 6 7 7 8 3 9 9 10 10 6	YES 4 4 5 9 10

Замечание

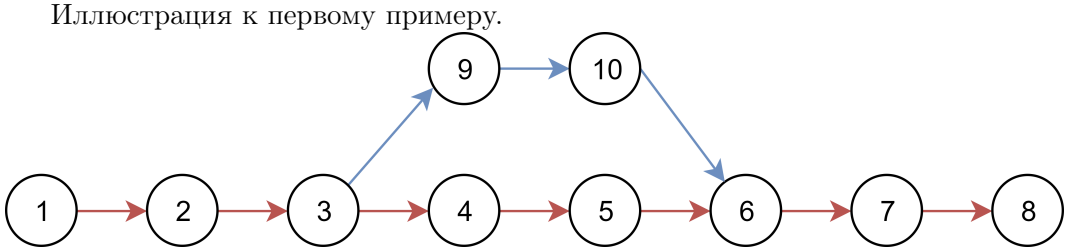



Иллюстрация к первому примеру.