

Problem A. Manhattan

Input file: *standard input*
Output file: *standard output*
Time limit: 1 second
Memory limit: 256 mebibytes

На Манхэттене для каждого целого i существуют улицы $x = i$ и $y = i$. Дом Snuke и дом Smeke расположены около улиц, при этом обычное (евклидово) расстояние между ними равно d .

Вычислите наибольшее возможное расстояние между их домами в случае, когда передвигаться разрешено только вдоль улиц.

Input

Входной файл содержит одно число d , такое, что $0 < d \leq 10$, а десятичная запись числа d содержит ровно три знака после десятичной точки.

Output

Выведите ответ с абсолютной или относительной точностью не хуже 10^{-9} .

Examples

standard input	standard output
1.000	2.000000000000
2.345	3.316330803765

Problem B. Dictionary

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 256 mebibytes

У Snuke есть словарь, который содержит n попарно различных слов s_1, \dots, s_n . Каждое слово состоит из строчных латинских букв. Слова отсортированы лексикографически, то есть $s_1 < \dots < s_n$.

К сожалению, некоторые буквы в словаре стёрлись и стали нечитаемыми. Эти буквы заменены символами '?'. Вычислите количество способов восстановить корректный словарь. Так как ответ может быть слишком большим, выведите остаток от деления ответа на $10^9 + 7$.

Input

Первая строка входа содержит одно целое число n ($1 \leq n \leq 50$). Далее следуют n строк, i -я из которых содержит слово s_i ($1 \leq |s_i| \leq 20$, s_i состоит из строчных английских букв или является '?').

Output

Выведите одно целое число — ответ к задаче.

Examples

standard input	standard output
2 ?sum??mer c??a??mp	703286064
3 snuje ????e snule	1

Problem C. Clique Coloring

Input file: *standard input*
Output file: *standard output*
Time limit: 1 second
Memory limit: 256 mebibytes

Дан полный граф с m вершинами. Изначально ни одно ребро графа не окрашено.

Snuke для каждого $i (1 \leq i \leq n)$ делает следующее действие: выбирает a_i вершин из графа и окрашивает все рёбра, которые соединяют какие-то две из выбранных вершин, цветом i . При этом ни одно ребро перекрашивать не пришлось.

При каком наименьшем значении m это возможно?

Input

Первая строка входа содержит одно целое число $n (1 \leq n \leq 5)$. Далее идут n строк, i -я из которых содержит одно целое число $a_i (2 \leq a_i \leq 10^9)$.

Output

Выведите наименьшее возможное значение m .

Examples

standard input	standard output
2 3 3	5
5 2 3 4 5 6	12

Note

В первом примере занумеруем вершины в графе как 1, 2, 3, 4, 5. Тогда можно раскрасить граф следующим образом:

- Выберем три вершины 1, 2, 3 и покрасим рёбра между ними в первый цвет.
- Выберем три вершины 1, 4, 5 и покрасим рёбра между ними во второй цвет..

Problem D. Dense Amidakuji

Input file: *standard input*
 Output file: *standard output*
 Time limit: 2 seconds
 Memory limit: 256 mebibytes

Amidakuji — известная японская игра. Игра содержит w (где w — чётное) длинных вертикальных отрезков и Snuke может добавить несколько коротких горизонтальных отрезков между ними. Каждый горизонтальный отрезок соединяет два соседних вертикальных отрезка. Всего есть h слоёв и каждый горизонтальный отрезок лежит в одном из слоёв. Таким образом, всего есть $h(w - 1)$ слотов для горизонтальных отрезков.

Пусть (a, b) — a -й сверху и b -й слева при нумерации с единицы слот (см. рисунок). Snuke сначала добавляет горизонтальные отрезки во все слоты (a, b) , соответствующие $a \equiv b \pmod{2}$. Затем он удаляет n горизонтальных отрезков в слотах $(a_1, b_1), \dots, (a_n, b_n)$.

Игра идёт следующим образом. Сначала Snuke выбирает i -й вертикальный отрезок. Затем он начинает движение с самого верха выбранного отрезка вниз. Когда он достигает какого-то горизонтального отрезка, он перемещается по этому отрезку и продолжает движение вниз уже по соседней вертикали. Игра прекращается, когда Snuke дойдёт до нижнего конца вертикального отрезка. Для каждого i вычислите номер вертикального отрезка, на котором Snuke финиширует.

Input

Первая строка входа содержит три целых числа h, w и n ($1 \leq h, w, n \leq 2 \cdot 10^5$, w чётно). Далее идут n строк; i -я из них содержит два целых числа a_i и b_i ($1 \leq a_i \leq h$, $1 \leq b_i \leq w - 1$, $a_i \equiv b_i \pmod{2}$, (a_i, b_i) попарно различны).

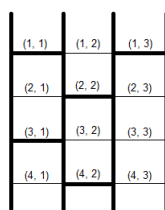
Output

Выведите w строк. В i -й строке выведите номер вертикального отрезка, в котором Snuke финиширует при старте на i -м отрезке.

Examples

standard input	standard output
4 4 1 3 3	2 3 4 1
10 6 10 10 4 4 4 5 1 4 2 7 3 1 3 2 4 8 2 7 5 7 1	1 4 3 2 5 6

Note



Если Snuke выберет самый левый вертикальный отрезок в первом примере, он будет перемещаться по слотам $(1, 1)$, $(2, 2)$, $(4, 2)$ и финиширует на втором слева отрезке

Problem G. Snake

Input file: *standard input*
Output file: *standard output*
Time limit: 1 second
Memory limit: 256 mebibytes

Назовём *змейкой* ломаную с n вершинами без самопересечений. Первоначально координаты i -й вершины змейки равны (x_i, y_i) , Змейка может непрерывно двигаться параллельным переносом и вращением, но она не может менять свою форму (то есть длины отрезков и углы между ними).

Прямая $y = 0$ является стеной, и в точке $(0, 0)$ есть отверстие. Проверьте, пройдёт ли змейка в отверстие; иначе говоря, первоначально y -координаты всех точек змейки были положительными; после движения эти координаты должны стать отрицательными.

Input

Первая строка входа содержит одно целое число n ($2 \leq n \leq 1000$). Далее следуют n строк, i 'я из которых содержит два целых числа x_i и y_i ($0 \leq x_i \leq 10^9$, $1 \leq y_i \leq 10^9$, $(x_i, y_i) \neq (x_{i+1}, y_{i+1})$). Ломаная не содержит самопересечений; никакие три точки ломаной не находятся на одной прямой.

Output

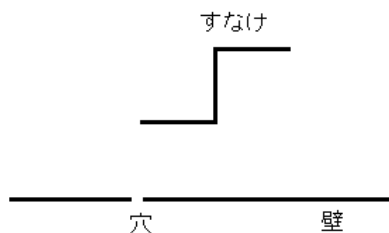
Если змейка может пройти сквозь отверстие, выведите "Possible". Иначе выведите "Impossible".

Examples

standard input	standard output
4 0 1 1 1 1 2 2 2	Possible
11 63 106 87 143 102 132 115 169 74 145 41 177 56 130 28 141 19 124 0 156 22 183	Impossible

Note

Для первого примера решение может выглядеть так:



- Сдвинуться на 1 в направлении $-y$.
- Повернуться на 90 градусов против часовой стрелки вокруг точки $(0, 0)$.
- Сдвинуться на 1 в направлении $-y$.
- Повернуться на 90 градусов по часовой стрелке вокруг точки $(0, 0)$.
- Сдвинуться на 1 в направлении $-y$.
- Повернуться на 90 градусов против часовой стрелки вокруг точки $(0, 0)$.
- Сдвинуться на 2 в направлении $-y$.

Problem J. Hyperrectangle

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 256 mebibytes

Snuke получил в подарок на день рождения d -мерный гиперпараллелепипед размером $l_1 \times \dots \times l_d$. Snuke разместил гиперпараллелепипед так, чтобы i -я координата его точек принимала значения от 0 до l_i и откусил кусок гиперпараллелепипеда, содержащий все точки, удовлетворяющие условию $x_1 + \dots + x_d \leq s$. (здесь x_i обозначает i -ю координату).

Пусть V — объём съеденной части. Можно показать, что $d!V$ (V умножить на факториал d) всегда является целым числом, если d_i и s также целые. Вычислите остаток от деления $d!V$ на $10^9 + 7$.

Input

Первая строка входа содержит одно целое число d ($2 \leq d \leq 300$). Далее следуют d строк, i -я из которых содержит одно целое число l_i ($1 \leq l_i \leq 300$). Последняя строка содержит одно целое число s ($0 \leq s \leq \sum l_i$).

Output

Выведите остаток от деления $d!V$ на $10^9 + 7$.

Examples

standard input	standard output
2 6 3 4	15
5 12 34 56 78 90 123	433127538

Note

Иллюстрация к первому примеру:



Problem K. Beads (Division 2 Only!)

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 256 mebibytes

Задано ожерелье из N жемчужин, пронумерованных последовательными целыми числами от 1 до N по часовой стрелке. Каждая жемчужина помечена заглавной латинской буквой. Таким образом, если прочитать эти буквы последовательно по часовой стрелке, получится строка длины N . Так как ожерелье замкнуто, то в зависимости от того, с какой жемчужины начинать чтение, можно получить различные строки.

Требуется найти среди этих строк лексикографически наименьшую.

Input

Первая строка входа содержит целое число N ($1 \leq N \leq 2 \cdot 10^6$) — длину ожерелья. Во второй строке задано само ожерелье. Ожерелье задаётся строкой, содержащей N заглавных латинских букв; k -я буква — это буква, написанная на k -й жемчужине.

Output

Выведите номер жемчужины, начиная с которой, можно прочитать по часовой стрелке лексикографически наименьшую строку длиной N . Если ответов несколько, выведите наименьший.

Example

standard input	standard output
6 CABCAB	2

Problem L. The Maximum Sum (Division 2 Only!)

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 256 mebibytes

Заданы N целых положительных чисел. Среди них выбираются два, сумма которых является наибольшей, не превосходящей заданного числа M . Требуется найти значение соответствующей суммы.

Input

Первая строка входа содержит два целых положительных числа N ($3 \leq N \leq 100$) и M ($1 \leq M \leq 100$). Вторая строка содержит N целых положительных чисел.

Output

Выведите одно число — требуемую сумму.

Examples

standard input	standard output
5 8 5 3 4 6 5	8
4 116 31 52 73 84	115

Problem M. Spellcheck (Division 2 Only!)

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 256 mebibytes

Вася написал спеллчекер, который выдаёт ошибки в следующих случаях:

- “u” или “ur” вместо “you” или “your”.
- “would of”, “should of” вместо “would have”, “should have”.
- “lol”. Более того, программа выдаёт ошибку на любое слово, в котором можно прочитать “lol” как подслово (если слово содержит “lol” несколько раз, как словл “lolololo”, всё равно выдаётся одна ошибка).

Напишите программу, которая считывает предложения одно за другим и для каждого предложения определяет, сколько ошибок найдёт новый спеллчекер.

Input

Первая строка входа содержит целое число T ($1 \leq T \leq 50$) — количество предложений в тесте. Каждая из последующих T строк содержит одно предложение — одно или несколько слов, разделённых пробелами. Слова состоят из строчных латинских букв. Слова разделены ровно одним пробелом, пробелы в начале и в конце предложения отсутствуют. Общая длина каждого предложения (вместе с пробелами) не превосходит 100 символов.

Output

Для каждого предложения в отдельной строке выведите, сколько ошибок найдёт в нём новый спеллчекер.

Examples

standard input	standard output
5	2
r u haz trololo	0
my name is vasya	1
i got the lollipop	3
u should of lolllollol	0
i should off line	

Problem N. Bluetooth (Division 2 Only)

Input file: *standard input*
Output file: *standard output*
Time limit: 2 seconds
Memory limit: 256 mebibytes

Вы пытаетесь отправить сообщение другу через Bluetooth. Смартфон может соединиться по Bluetooth с любым смартфоном, если расстояние между ними не превышает D .

Определите, можно ли отправить сообщение «по цепочке» (напрямую или через некоторое количество промежуточных смартфонов).

Input

Первая строка входа содержит два целых числа N ($1 \leq N \leq 10$) и D ($1 \leq D \leq 10$), где N — общее количество смартфонов с Bluetooth в помещении, а D — максимальное расстояние, на котором два смартфона соединяются по Bluetooth.

Каждая из следующих N строк содержит по два целых числа — координаты X и Y очередного смартфона. При этом первая строка задаёт координаты Вашего смартфона, последняя — координаты смартфона Вашего друга ($0 \leq X, Y \leq 100$).

Output

Выведите 'y', если сообщение передать можно и 'n' в противном случае.

Example

standard input	standard output
4 7 1 4 6 2 9 7 14 4	y
5 6 7 1 5 5 1 6 8 7 20 15	n