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 $s_1, \ldots, s_n$. Каждое слово состоит из строчных латинских букв. Слова отсортированы лексикографически, то есть $s_1 < \cdots < s_n$.

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 ($1 \le n \le 50$). Далее следуют n строк, i-я из которых содержит слово s_i ($1 \le |s_i| \le 20$, 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 $i(1 \le i \le n)$ 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 $(1 \le n \le 5)$. Далее идут n строк, i-я из которых содержит одно целое число a_i $(2 \le a_i \le 10^9)$.

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 $(a_1, b_1), \ldots, (a_n, b_n)$.

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

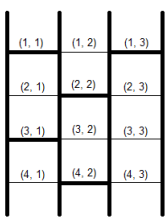
Output

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 E. Cellular Automaton

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

Пусть w — целое положительное число и p — строка длины 2^{2w+1}. $(w, p)-$ является конечным автоматом, определённым следующим образом.

- Ячейки соединены в линию и образуют бесконечно длинную цепочку.
- Каждая ячейка может принимать значения 0 и 1.
- В момент времени 0 Snuke выбирает некоторый конечный набор ячеек и устанавливает их значения в 1. Остальные ячейки содержат нули.
- Пусть $f(t, x)$ — состояние ячейки x в момент времени $t(> 0)$. $f(t, x)$ определяется из $f(t-1, x-w), \cdots, f(t-1, x+w)$ в соответствии со следующим правилом:

$$f(t, x) = p[\sum_{i=-w}^{w} 2^{w+i} f(t-1, x+i)] \quad (1)$$

Автомат нравится Snuke, если количество единиц не меняется вне зависимости от того, какой выбор был сделан в момент времени 0. По заданным w и строке s найдите лексикографически наименьшее p такое, что $s \leq p$ и $(w, p)-$ автомат нравится Snuke.

Input

Первая строка входа содержит одно целое число w ($1 \leq w \leq 3$). Вторая строка содержит s ($|s| = 2^{2w+1}$, s consists of '0' and '1'.

Output

Выведите наименьшую возможную строку p. Если таких строк нет, выведите "no".

Examples

standard input	standard output
1 00011000	00011101
1 11111111	no

Problem F. Directions

Input file:	*standard input*
Output file:	*standard output*
Time limit:	4 seconds
Memory limit:	256 mebibytes

Snuke собирается путешествовать. Изначально Snuke не может никуда поехать. Есть n билетов, цена i-го билета равна p_i. Если Snuke покупает i-й билет, он может для всех точек (x, y) и неотрицательного числа t ехать из (x, y) в $(x + ta_i, y + tb_i)$. Snuke собирается купить билеты таким образом, чтобы он мог путешествовать между любыми двумя точками. Какую минимальную сумму он при этом потратит?

Input

Первая строка ввода содержит одно целое число n ($1 \le n \le 2 \cdot 10^5$). Далее следуют n строк; в i-й из этих строк содержит три целых числа a_i, b_i, p_i ($-10^9 \le a_i, b_i \le 10^9$, $1 \le p_i \le 10^9$).

Output

Выведите минимальную цену, которую Snuke должен заплатить за билеты для того, чтобы он мог передвигаться между двумя любыми точками. Если он не сможет это сделать даже при покупке всех билетов, выведите -1.

Examples

standard input	standard output
7 0 3 1 0 3 2 1 -1 2 0 0 1 -2 4 1 -4 0 1 2 1 2	4
2 1 2 3 4 5 6	-1

Note

В первом тестовом примере Snuke может выбрать, например, билеты 1, 3 и 6.

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 положительны; после движения эти координаты должны стать отрицательными.

Input

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

Output

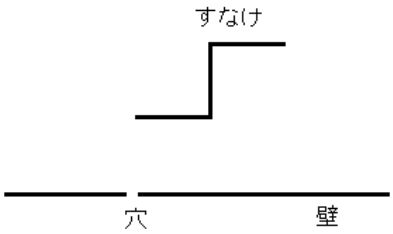
Если змейка может пройти сквозь отверстие, выведите "`Possible`". Иначе выведите "`Impossible`".

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 H. Distance Sum

Input file:	*standard input*
Output file:	*standard output*
Time limit:	4 seconds
Memory limit:	256 mebibytes

На некоторой карте обозначены n городов и $n-1$ дорога, соединяющая эти города таким образом, что полученный граф является деревом. Города занумерованы последовательными целыми числами от 1 до n.

Город 1 является корнем дерева; обозначим для каждого $i > 1$ город, являющийся предком города i, за p_i, а расстояние между городами p_i и i за d_i.

Snuke хочет для каждого $1 \le k \le n$ вычислить наименьшую сумму расстояний от некоторого города до городов $1, \ldots, k$:

$$\min_{1 \le v \le n} \{\sum_{i=1}^{k} dist(i, v)\} \tag{2}$$

Здесь $dist(u, v)$ обозначает расстояние между городами u и v.

Input

Первая строка входа содержит одно целое число n ($1 \le n \le 2 \cdot 10^5$). Далее идут $n-1$ строк, i-я из которых содержит два целых числа p_{i+1} и d_{i+1} — номер предка города $i+1$ и расстояние между городом $i+1$ и этим предком ($1 \le p_i \le n$, $1 \le d_i \le 2 \cdot 10^5$, p_i образуют дерево).

Output

Выведите n строк. В i-й из этих строк выведите ответ для $k = i$.

Examples

standard input	standard output
10 4 1 1 1 3 1 3 1 5 1 6 1 6 1 8 1 4 1	0 3 3 4 5 7 10 13 16 19
15 1 3 12 5 5 2 12 1 7 5 5 1 6 1 12 1 11 1 12 4 1 1 5 5 10 4 1 2	0 3 9 13 14 21 22 29 31 37 41 41 47 56 59

Problem I. Substring Pairs

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

Snuke нашёл несколько интересных пар строк (s, t), однако он забыл сами строки. Впрочем, он помнит про них следующее:

- Длина строки s в точности равна N.
- Длина строки t в точности равна M.
- t является подстрокой s. (то есть можно выбрать M последовательных символов в s так, что получившаяся строка совпадёт с t.)

Вычислите количество требуемых пар строк $((s, t))$ по модулю $10^9 + 7$, если алфавит состоит из A символов.

Input

Первая строка входа содержит три целых числа N, M и A ($1 \le N \le 200$, $1 \le M \le 50$, $M \le N$, $1 \le A \le 1000$)

Output

Выведите одно число — остаток от деления количества пар строк (s, t), удовлетворяющих соотношениям, заданным выше на $10^9 + 7$.

Examples

standard input	standard output
3 2 2	14
200 50 1000	678200960

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 $l_1 \times \cdots \times l_d$. 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 $x_1 + \cdots + x_d \leq s$. (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:

