Задача A. Наташа, Саша и префиксные суммы. Перерождение.

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

Два любимых числа Наташи — это n и 1, а два любимых числа Саши — это $-m$ и -1. Однажды Саша и Наташа встретились и выписали все возможные массивы из $n+m$ элементов, в которых n элементов равны 1, а другие m элементов равны -1. Для каждого массива они посчитали его максимальную префиксную сумму, возможно пустую, равную 0 (т.е. если какждая непустая префиксная сумма меньше нуля, то она считается равной нулю). Более формально, обозначим за $f(a)$ максимальную префиксную сумму массива $a_{1,\ldots,l}$ длины $l \geqslant 0$. Тогда:

$$f(a) = max(0, \max_{1 \leqslant i \leqslant l} \sum_{j=1}^{i} a_j)$$

Теперь они хотят посчитать сумму максимальных префиксных сумм по всем выписанным массивам и просят вас в этом помочь. Так как эта сумма может быть очень большой, выведите ее по модулю 998244853.

Формат входных данных

Единственная строка содержит два целых числа n и m ($0 \leqslant n, m \leqslant 10^6$).

Формат выходных данных

Выведите ответ по модулю 998244853.

Примеры

стандартный ввод	стандартный вывод
0 2	0
2 0	2
2 2	5
2000 2000	674532367

Замечание

В первом примере существует единственный возможный массив: $[-1, -1]$, его максимальная префиксная сумма равна 0.

Во втором примере существует единственный возможный массив: $[1, 1]$, его максимальная префиксная сумма равна 2.

В третьем примере существуют 6 возможных массивов:

- $f([1, 1, -1, -1]) = 2$
- $f([1, -1, 1, -1]) = 1$
- $f([1, -1, -1, 1]) = 1$
- $f([-1, 1, 1, -1]) = 1$
- $f([-1, 1, -1, 1]) = 0$
- $f([-1, -1, 1, 1]) = 0$

Ответ для третьего примера равен $2 + 1 + 1 + 1 + 0 + 0 = 5$.

Задача B. Черепашки

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

Дана таблица размера $n \times m$. Будем считать, что строки таблицы пронумерованы сверху вниз от 1 до n, а столбцы — слева направо от 1 до m. Тогда клетку, находящуюся в строке x и столбце y, будем обозначать (x, y).

Изначально в клетке $(1, 1)$ стоят две одинаковые черепашки. Обе черепашки хотят попасть в клетку (n, m). В некоторых клетках таблицы существуют препятствия, но гарантируется, что в левом верхнем и правом нижнем углах их нет. Из клетки (x, y) черепашка (как одна, так и другая) может перейти в одну из двух клеток $(x+1, y)$ и $(x, y+1)$, если соответствующая клетка не содержит препятствия. Черепашки поссорились, поэтому они не хотят, чтобы на протяжении пути была хоть какая-то вероятность встретиться. Помогите им найти количество способов, которыми они могут пройти от клетки $(1, 1)$ до клетки (n, m).

Более формально, найдите количество пар непересекающихся путей из клетки $(1, 1)$ в клетку (n, m) по модулю 1000000007 ($10^9 + 7$). Два пути называются непересекающимися, если у них есть ровно две общие клетки — начальная и конечная.

Формат входных данных

В первой строке записаны два целых числа n, m ($2 \leqslant n, m \leqslant 3000$). В каждой из следующих n строк записано m символов — описание таблицы. Свободные клетки обозначаются символом «`.`», клетки с препятствиями — символом «`#`».

Гарантируется, что левая верхняя и правая нижняя клетки свободны.

Формат выходных данных

В единственной строке выведите целое число — количество пар непересекающихся путей из клетки $(1, 1)$ в клетку (n, m) по модулю 1000000007 ($10^9 + 7$).

Примеры

стандартный ввод	стандартный вывод
4 5 ..... .###. .###.	1
2 3	1

Задача C. Число остовных деревьев

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

Дан связный неориентированный граф. Найдите число его остовных деревьев по модулю 10^9+7.

Формат входных данных

Первая строка входного файла содержит натуральные числа n и m — количество вершин и ребер графа ($1 \leqslant n \leqslant 100$, $1 \leqslant m \leqslant 1\,000$). Далее следуют m строк, задающих ребра. Граф не содержит кратных ребер и петель.

Формат выходных данных

Выведите число остовных деревьев по модулю 10^9+7.

Примеры

стандартный ввод	стандартный вывод
4 4 2 1 4 2 2 3 4 3	3

Задача D. Линейные уравнения

Имя входного файла:	`linear.in`
Имя выходного файла:	`linear.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Система линейных уравнений, как всем известно, есть множество уравнений

$$\begin{aligned} a_{11}x_1 + \ldots + a_{1n}x_n &= b_1 \\ &\ldots \\ a_{n1}x_1 + \ldots + a_{nn}x_n &= b_n \end{aligned}$$

Ваша задача — решить её.

Формат входных данных

В первой строке входного файла записано целое число n ($1 \leqslant n \leqslant 20$). В следующих n строках записано по $n+1$ целых чисел: $a_{i1}, \ldots, a_{in}, b_i$. Все эти числа не превышают 100 по абсолютному значению.

Формат выходных данных

Первая строка выходного файла должна содержать одно из следующих сообщений:

- `impossible` — решений нет
- `infinity` — бесконечно много решений
- `single` — единственное решение. В этом случае вторая строка должна содержать n чисел $x_1, \ldots, x_n$, разделенных пробелами. Решение должно быть выведено с точностью не менее трех знаков после десятичной точки.

Примеры

linear.in	linear.out
2 1 1 1 2 2 2	infinity
2 1 2 0 1 2 1	impossible

Задача E. Электричество в каждый дом!

Имя входного файла:	`countspans.in`
Имя выходного файла:	`countspans.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Известный чешский математик Отакар Борувка крайне увлечен проектированием электросети Моравии. В Моравии N городов, и различные строительные фирмы уже предложили Отакару M проектов построения ЛЭП между какими-то двумя городами. Известно, что в погоне за индивидуальностью и неповторимым строительным почерком, каждая фирма предлагает все свои проекты по одинаковой стоимости, отличной от стоимостей проектов других фирм. Также известно, что каждая фирма предлагает не более *трех* проектов. Можете считать, что фирмы достаточно сообразительны, чтобы не предлагать проектов соединения какого то города с самим собой, но вполне может возникнуть ситуация, что одна или несколько фирм предлагают больше одного проекта соединения одной и той же пары городов.

Борувка собрал всех своих друзей и поручил им задачу спроектировать электросеть минимальной стоимости. Как вы, наверное, уже догадались, электрическая сеть является остовным деревом, а Борувку интересуют только сети, стоимость постройки которых минимальна.

Борувка всегда был уверен, что минимальное остовное дерево у графа одно, и представьте себе его удивление, когда каждый из друзей принес ему свой проект, утверждая что его-то дерево и есть минимальное. Подозревая неладное, он думает, что причиной разных ответов стали ребра одинакового веса. Помогите ему — посчитайте количество возможных электросетей минимальной стоимости, состоящих из ЛЭП, предложенных Борувке.

Формат входных данных

В первой строке входного файла заданы два целых положительных числа N и M, не превосходящие 100 000.

Следующие M строк содержат по 3 целых числа каждая: $1 \leqslant a_i, b_i \leqslant N$ и $1 \leqslant c_i \leqslant 10^9$ — города, соединенные соответствующей ЛЭП, и ее стоимость. Гарантируется, что для любого числа c_i найдется не более трех ЛЭП, имеющих такую стоимость.

Формат выходных данных

Выведите одно число — ответ на задачу. Так как это число может оказаться довольно большим, выведите остаток от деления на $10^9 + 7$.

Примеры

countspans.in	countspans.out
2 2 1 2 1 2 1 1	2
3 5 1 2 2 2 3 3 3 1 3 3 1 1 2 3 1	1
5 10 2 5 8 4 5 8 2 4 3 4 2 3 4 2 3 5 4 6 5 1 6 1 3 5 3 1 5 4 5 7	6

Задача F. Совершенство

Имя входного файла:	`matchme.in`
Имя выходного файла:	`matchme.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Павел Андреевич очень любит все совершенное. Его новая любовь - совершенные паросочетания! Не огорчайте Павла Андреевича, проверьте, есть ли в данном графе совершенное паросочетание.

Формат входных данных

В первой строке даны два числа N и M - количество вершин и количество ребер в G. $N \leqslant 100$. Следующие M строк содержат числа a_i и b_i - ребра графа. Гарантируется отсутствие петель и кратных ребер.

Формат выходных данных

Выведите слово «YES» или слово «NO» - ответ на поставленную задачу.

Примеры

matchme.in	matchme.out
6 7 1 2 2 3 1 3 5 6 6 4 4 5 6 2	YES
3 3 1 2 2 3 1 3	NO

Задача G. Максимизировать сумму XOR

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	3 секунды
Ограничение по памяти:	512 мегабайт

Будем обозначать как $\oplus$ операцию *побитового «исключающего или»* для целых чисел. В языках программирования C++ и Java она обозначается символом «`^`», в паскале и Python — ключевым словом «`xor`». Например, $9 \oplus 3 = 1001_2 \oplus 11_2 = 1010_2 = 10$.

Даны два массива A и B длины n. Обозначим как $X(A)$ для массива A результат вычисления побитового «исключающего или» от всех элеметов массива: $X(A) = A_1 \oplus A_2 \oplus \ldots \oplus A_n$. Аналогично, введем обозначение $X(B) = B_1 \oplus B_2 \oplus \ldots \oplus B_n$.

Для каждого i от 1 до n разрешается поменять местами элементы A_i и B_i. Необходимо определить, какие из этих обменов надо сделать, чтобы максимизировать сумму $X(A) + X(B)$.

Формат входных данных

В первой строке входных данных находится число n — количество элементов ($1 \leqslant n \leqslant 10^5$). В следующей строке находится n элементов массива A ($0 \leqslant A_i \leqslant 10^{18}$). В следующей строке в таком же формате дан массив B.

Формат выходных данных

В первой строке выведите максимальную возможную сумму и число k — количество необходимых обменов. В следующей строке выведите k различных чисел от 1 до n — индексы элементов, которые надо поменять.

Пример

стандартный ввод	стандартный вывод
2 1 1 2 2	6 1 1

Замечание

В примере после обмена массивы равны $A = [2, 1]$ и $B = [1, 2]$, соответственно.
$X(A) = 2 \oplus 1 = 10_2 \oplus 1_2 = 11_2 = 3$, $X(B) = 3$, $X(A) + X(B) = 6$.

Задача H. Рандомные роботы

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

На прямой стоят K роботов. i-й робот изначально расположен в координате x_i.
Повторим следующую процедуру N раз:

- Каждый робот двигается с вероятностью $\frac{1}{2}$. В случае, если робот двигается, то его x координата увеличивается, иначе он остается на месте. На i-м шаге все движения происходят одновременно.

Найдите вероятность того, что никакие два робота не окажутся в одной и той же координате одновременно в течение выполнения процесса. Выведите ответ по модулю 998244353.

Формат входных данных

В первой строке входных данных содержится пара целых чисел K, N ($1 \leqslant K \leqslant 10, 1 \leqslant N \leqslant 1000$) — количество роботов и длительность процесса соответственно.

В следующей строке содержатся K целых чисел x_i ($0 \leqslant x_i \leqslant 1000$) — изначальные координаты роботов.

Формат выходных данных

В единственной строке выходных данных выведите ответ на задачу по модулю 998244353.

Примеры

стандартный ввод	стандартный вывод
2 2 1 2	374341633
2 2 10 100	1
10 832 73 160 221 340 447 574 720 742 782 970	553220346

Задача I. Дерево + дерево

Имя входного файла:	`стандартный ввод`
Имя выходного файла:	`стандартный вывод`
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	256 мегабайт

Дан граф G из N вершин, пронумерованных от 1 до N. Изначально в графе 0 ребер. Вам даны две последовательности $(u_1, u_2, \ldots, u_M)$ и $(v_1, v_2, \ldots, v_M)$ длины M.

Вы выполняете следующую операцию $N-1$ раз:

- Выберите случайное i $(1 \leqslant i \leqslant M)$. Добавьте в G ребро, соединяющее вершины u_i и v_i.

Обратите внимание на то, что данная операция добавит ребро между вершинами, даже если оно уже присутствует в графе. Иными словами, в графе G могут быть кратные ребра.

Для каждого $K = 1, 2, \ldots, N-1$ найдите вероятность того, что G будет лесом после выполнения K операций по модулю 998244353.

Формат входных данных

В первой строке содержатся два целых числа N, M. $(2 \leqslant N \leqslant 14, N-1 \leqslant M \leqslant 500)$ В следующих M строках содержатся пары целых чисел $(1 \leqslant u_i, v_i \leqslant N)$. Гарантируется, что $u_i \neq v_i$.

Формат выходных данных

Выведите $N-1$ строку. В i-е строке должна содержатся вероятность того, что G является лесом после i операций по модулю 998244353.

Примеры

стандартный ввод	стандартный вывод
3 2 1 2 2 3	1 499122177
4 5 1 2 1 2 1 4 2 3 2 4	1 758665709 918384805