Задача A. Обход в глубину

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

Дан неориентированный невзвешенный граф, в котором выделена вершина. Вам необходимо найти количество вершин, лежащих с ней в одной компоненте связности (включая саму выделенную вершину).

Формат входных данных

В первой строке входного файла содержатся два целых числа N и S ($1 \leqslant S \leqslant N \leqslant 100$), где N — количество вершин графа, а S — выделенная вершина. В следующих N строках записано по N чисел — матрица смежности графа, в которой цифра «0» означает отсутствие ребра между вершинами, а цифра «1» — его наличие. Гарантируется, что на главной диагонали матрицы всегда стоят нули.

Формат выходных данных

Выведите одно целое число — искомое количество вершин.

Примеры

dfs.in	dfs.out
5 1 0 1 1 0 0 1 0 1 0 0 1 1 0 0 0 0 0 0 0 1 0 0 0 1 0	3

Задача B. Количество достижимых вершин

Имя входного файла:	`reach.in`
Имя выходного файла:	`reach.out`
Ограничение по времени:	1 секунда
Ограничение по памяти:	64 мегабайта

Вам дан ориентированный граф, заданный списками смежности. Он состоит из N вершин. Посчитайте количество вершин, достижимых из вершины с номером S. Любая вершина считается достижимой из самой себя.

Формат входных данных

В первой строке входного файла записано два числа N и S ($1 \leqslant N \leqslant 10^4$, $1 \leqslant S \leqslant N$). Далее идут N строк. В i-й строке содержится количество вершин, смежных с вершиной i, и номера этих вершин. Все вершины нумеруются натуральными числами от 1 до N. Количество рёбер в графе не превышает 10^5.

Формат выходных данных

Выведите в выходной файл одно число — количество вершин, достижимых из вершины с номером S.

Примеры

reach.in	reach.out
4 1 1 4 2 1 4 2 1 4 1 3	3

Задача C. Компоненты связности

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

Дан неориентированный невзвешенный граф. Необходимо посчитать количество его компонент связности.

Формат входных данных

В первой строке входного файла содержится одно натуральное число N ($N \leqslant 100$) — количество вершин в графе. Далее в N строках по N чисел — матрица смежности графа: в i-й строке на j-м месте стоит «`1`», если вершины i и j соединены ребром, и «`0`», если ребра между ними нет. На главной диагонали матрицы стоят нули. Матрица симметрична относительно главной диагонали.

Формат выходных данных

Вывести одно целое число — искомое количество компонент связности графа.

Примеры

components.in	components.out
6 0 1 1 0 0 0 1 0 1 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0	3

Задача D. Есть ли цикл?

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

Дан ориентированный граф. Требуется определить, есть ли в нем цикл.

Формат входных данных

В первой строке вводится натурально число N ($N \leqslant 50$) — количество вершин. Далее в N строках следуют по N чисел, каждое из которых — «0» или «1». j-е число в i-й строке равно «1» тогда и только тогда, когда существует ребро, идущее из i-й вершины в j-ю. Гарантируется, что на диагонали матрицы будут стоять нули.

Формат выходных данных

Выведите «0», если в заданном графе цикла нет, и «1», если он есть.

Примеры

cycle.in	cycle.out
3 0 1 0 0 0 1 0 0 0	0

Задача E. Долой списывание!

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

Во время теста Михаил Дмитриевич заметил, что некоторые лкшата обмениваются записками. Сначала он хотел поставить им всем двойки, но в тот день Михаил Дмитриевич был добрым, а потому решил разделить лкшат на две группы: списывающих и дающих списывать, и поставить двойки только первым.

У Михаила Дмитриевича записаны все пары лкшат, обменявшихся записками. Требуется определить, сможет ли он разделить лкшат на две группы так, чтобы любой обмен записками осуществлялся от лкшонка одной группы лкшонку другой группы.

Формат входных данных

В первой строке находятся два числа N и M — количество лкшат и количество пар лкшат, обменивающихся записками ($1 \leqslant N \leqslant 100$, $0 \leqslant M \leqslant \frac{N(N-1)}{2}$). Далее в M строках расположены описания пар лкшат: два различных числа, соответствующие номерам лкшат, обменивающихся записками (нумерация лкшат идёт с 1). Каждая пара лкшат перечислена не более одного раза.

Формат выходных данных

Необходимо вывести ответ на задачу Павла Олеговича. Если возможно разделить лкшат на две группы, выведите «`YES`»; иначе выведите «`NO`».

Примеры

bipartite.in	bipartite.out
3 2 1 2 2 3	YES

Задача F. Поиск цикла

Имя входного файла:	`cycle.in`
Имя выходного файла:	`cycle.out`
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайта

Дан ориентированный невзвешенный граф. Необходимо определить есть ли в нём циклы, и если есть, то вывести любой из них.

Формат входных данных

В первой строке входного файла находятся два натуральных числа N и M ($1 \leqslant N \leqslant 100\,000$, $M \leqslant 100\,000$) — количество вершин и рёбер в графе соответственно. Далее в M строках перечислены рёбра графа. Каждое ребро задаётся парой чисел — номерами начальной и конечной вершин соответственно.

Формат выходных данных

Если в графе нет цикла, то вывести «`NO`», иначе — «`YES`» и затем перечислить все вершины в порядке обхода цикла.

Примеры

cycle.in	cycle.out
2 2 1 2 2 1	YES 2 1
2 2 1 2 1 2	NO