

Задача 1 Сравнение

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

Формат входного файла

Во входном файле даны два неотрицательных числа A и B ($0 \leq A, B \leq 10^{10\,000}$) каждое с своей строке.

Формат выходного файла

В выходной файл выведите «<», если $A < B$; «=», если $A = B$ и «>», если $A > B$.

Примеры

<code>compare.in</code>	<code>compare.out</code>
2	>
1	

Задача 2 Поиск цикла

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

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

Формат входного файла

В первой строке входного файла находятся два натуральных числа N и M ($1 \leq N \leq 100\,000$, $M \leq 100\,000$) — 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 выходного файла

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

Примеры

<code>cycle.in</code>	<code>cycle.out</code>
2 2 1 2 2 1	YES 1 2
2 2 1 2 1 2	NO

Задача 4 Выход из лабиринта

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

Во время торнадо в Костромской области было повалено большое количество деревьев. Однажды утром, спустя пару дней после торнадо, мальчик Илья вышел прогуляться и увидел, что поваленные деревья образовали настоящий лабиринт. Вася очень обрадовался, стал гулять и играть в лабиринте. Внезапно ему позвонил преподаватель и поинтересовался, почему он давно не видел Илью. Чтобы не получить дырку в бэйджик за самовольный выход за территорию лагеря, Илья, конечно же, хотел попасть в домик как можно скорее, но и побродить по лабиринту ему тоже очень хотелось. Помогите ему узнать, как быстро он сможет добраться до домика.

Формат входного файла

В первой строке входного файла заданы целые положительные числа N и M ($1 \leq N, M \leq 1000$). В следующих N строках заданы по M символов, описывающих лабиринт. На позициях, по которым Илья может перемещаться, записан символ «0», а на позициях, которые перегородены деревьями, записан символ «1». После описания лабиринта следуют целые числа x_1, y_1, x_2, y_2 — координаты (то есть соответствующие номера столбцов и строк) Ильи и его домика соответственно ($1 \leq x_1, x_2 \leq M, 1 \leq y_1, y_2 \leq N$).

Формат выходного файла

В выходной файл выведите единственное число — длину кратчайшего пути от точки (x_1, y_1) до точки (x_2, y_2) , если Илья может добраться до domu по лабиринту, и «-1» в противном случае.

3 Firesafe. Противопожарная безопасность

Имя входного файла: `firesafe.in`
Имя выходного файла: `firesafe.out`

В Марксе n домов. Некоторые из них соединены дорогами с односторонним движением.

В последнее время в Марксе участились случаи пожаров. В связи с этим жители решили построить в городе несколько пожарных станций. Но возникла проблема: едущая по вызову пожарная машина, конечно, может игнорировать направление движения текущей дороги, однако, возвращаясь с задания машина обязана следовать правилам дорожного движения (жители Маркса свято чтут эти правила!).

Ясно, что где бы ни оказалась пожарная машина, у неё должна быть возможность вернуться на ту пожарную станцию, с которой выехала. Но строительство станций стоит больших денег, поэтому на совете города было решено построить минимальное количество станций таким образом, чтобы это условие выполнялось. Кроме того, для экономии было решено строить станции в виде пристроек к уже существующим домам.

Ваша задача — написать программу, рассчитывающую оптимальное положение станций.

Формат входного файла

В первой строке входного файла задано число n ($1 \leq n \leq 3\,000$). Во второй строке записано количество дорог m ($1 \leq m \leq 100\,000$). Далее следует описание дорог в формате $a_i\ b_i$, означающее, что по i -й дороге разрешается движение автотранспорта от дома a_i к дому b_i ($1 \leq a_i, b_i \leq n$).

Формат выходного файла

В первой строке выведите минимальное количество пожарных станций K , которые необходимо построить. Во второй строке выведите K чисел в произвольном порядке — дома, к которым необходимо пристроить станции. Если оптимальных решений несколько, выведите любое.

Пример

<code>firesafe.in</code>	<code>firesafe.out</code>
5 7 1 2 2 3 3 1 2 1 2 3 3 4 2 5	2 4 5

Примеры

<code>maze.in</code>	<code>maze.out</code>
4 6 1 0 1 1 1 0 0 0 1 0 0 0 1 0 1 1 0 0 0 0 0 0 0 0 2 1 5 3	7
4 6 1 0 1 1 1 0 0 0 1 0 0 0 1 0 1 1 0 0 0 0 0 1 0 0 2 1 5 3	-1
5 5 0 0 0 0 0 0 1 1 1 0 0 1 1 1 0 0 1 1 1 0 0 0 0 0 0 1 3 1 3	0

Задача 5 МегаНОД

Дано N чисел. Найти самое большое число, на которое делятся все N чисел. Циклы не использовать.

Формат входных данных

В первой строке дано число N . Во второй строке даны через пробел N чисел ($1 \leq N \leq 1000$).

Формат выходных данных

Выведите искомое число.

Примеры

входные данные	выходные данные
1 9	9

Задача 7 Истоки и стоки

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

Вершина ориентированного графа называется истоком, если в нее не входит ни одно ребро и стоком, если из нее не выходит ни одного ребра.

Ориентированный граф задан матрицей смежности. Найдите все его вершины-истоки и все вершины-стоки.

Формат входного файла

N — число вершин в графе ($1 \leq N \leq 100$), затем матрица смежности — N строк по N чисел, каждое из которых равно 0 или 1.

Формат выходного файла

В первой строке выведите K — число истоков в графе, затем номера вершин, являющихся истоками в порядке возрастания. Во второй строке выведите информацию о стоках в том же порядке.

Примеры

source.in	source.out
5 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0	2 3 4 3 1 4 5

Задача 6 Произведение

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

Формат входного файла

Во входном файле два целых неотрицательных числа A и B ($0 \leq A, B \leq 10^{10\,000}$) каждое в своей строке.

Формат выходного файла

В выходной файл выведите одно число, равное произведению A и B .

Примеры

multiply.in	multiply.out
65536 216	14155776

Задача 8 TopSort

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

Дан ориентированный невзвешенный граф. Необходимо его топологически отсортировать.

Формат входного файла

В первой строке входного файла два натуральных числа N и M ($1 \leq N \leq 10^5, 1 \leq M \leq 10^5$) — количество вершин и рёбер в графе соответственно. Далее в M строках перечислены рёбра графа. Каждое ребро задается парой чисел — номерами начальной и конечной вершин соответственно.

Формат выходного файла

Вывести любую топологическую сортировку графа в виде последовательности номеров вершин. Если граф невозможно топологически отсортировать, вывести -1.

Примеры

topsort.in	topsort.out
6 6 1 2 3 2 4 2 2 5 6 5 4 6	4 6 3 1 2 5
3 3 1 2 2 3 3 1	-1