Задача A. Обрати меня!

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

Мальчик Вася очень любит разворачивать ориентированные графы. Помогите ему в этом.

Формат входных данных

Во входном файле записано число N ($1 \leqslant N \leqslant 50\,000$) — количество вершин в графе. В следующих N строках записан граф в виде списков смежности: в i-й строке, начале записано количество вершин, смежных с вершиной под номером i, затем в порядке возрастания записаны номера вершин, в которые идут рёбра из i-й вершины. Нумерация начинается с единицы. Гарантируется, что рёбер в графе не более 50 000.

Формат выходных данных

Выведите развёрнутый граф в том же формате, что и исходный.

Примеры

reverse.in	reverse.out
4 2 2 3 1 3 0 1 2	4 0 2 1 4 2 1 2 0
2 1 2 1 1	2 1 2 1 1

Задача B. Лесопосадки

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

Дан неориентированный невзвешенный граф. Необходимо определить, является ли он деревом.

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-ой строке на j-ом 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 «`YES`», если граф является деревом, «`NO`» иначе.

Примеры

tree.in	tree.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	NO
3 0 1 0 1 0 1 0 1 0	YES

Задача C. Кратчайшее расстояние

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

Дан ориентированный граф. Найдите расстояния от вершины x до всех остальных вершин графа.

Формат входных данных

В первой строке входного файла содержатся два натуральных числа N и x ($1 \leqslant N \leqslant 1\,000$, $1 \leqslant x \leqslant N$) — количество вершин в графе и стартовая вершина соответственно. 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.

Формат выходных данных

Выведите через пробел числа $d_1, d_2, \ldots d_n$, где d_i — это -1, если путей между x и i нет, и минимальное расстояние между x и i в противном случае.

Примеры

mindist.in	mindist.out
6 5 0 1 1 0 0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1 0 0 0 1 0 0 0 0	2 2 1 1 0 -1

Задача D. Шайтан-машинка

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

У Ибрагима есть магическая чёрная шайтан-машинка. На ней есть три кнопки и табло. Табло может показывать не более чем четырёхзначные числа. Каждая из кнопок меняет число некоторым образом: первая домножает его на 3, вторая прибавляет к нему сумму его цифр, а третья вычитает из него 2. В случае, если число становится отрицательным или превосходит 9 999, шайтан-машинка ломается. Ибрагим может нажимать кнопки в любом порядке. Он хочет узнать, как получить на табло число b после некоторой последовательности нажатий, если сейчас шайтан-машинка показывает a. Помогите ему найти минимальное необходимое число нажатий.

Формат входных данных

Единственная строка входного файла содержит два натуральных числа a и b, разделённых пробелом ($1 \leqslant a, b \leqslant 9\,999$).

Формат выходных данных

Выведите одно число — минимальное необходимое количество действий.

Примеры

crazycalc.in	crazycalc.out
14 45	3
18 12	3
14 29	2

Задача E. Детали

Имя входного файла:	`details.in`
Имя выходного файла:	`details.out`
Ограничение по времени:	4 секунды
Ограничение по памяти:	64 мегабайта

Предприятие «Авто-2005» выпускает двигатели для известных во всём мире автомобилей. Двигатель состоит ровно из n деталей, пронумерованных от 1 до n, при этом деталь с номером i изготавливается за p_i секунд. Специфика предприятия «Авто-2005» заключается в том, что там одновременно может изготавливаться лишь одна деталь двигателя. Для производства некоторых деталей необходимо иметь предварительно изготовленный набор других деталей.

Генеральный директор «Авто-2005» поставил перед предприятием амбициозную задачу — за наименьшее время изготовить деталь с номером 1, чтобы представить её на выставке.

Требуется написать программу, которая по заданным зависимостям порядка производства между деталями найдёт наименьшее время, за которое можно произвести деталь с номером 1.

Формат входных данных

Первая строка написано натуральное число n ($1 \leqslant n \leqslant 100\,000$). Вторая — n натуральных чисел $p_1, p_2, \ldots, p_n$, определяющих время изготовления каждой детали в секундах ($1 \leqslant p_i \leqslant 1\,000\,000\,000$).

Каждая из последующих n строк входного файла описывает характеристики производства деталей. Здесь i-я строка содержит в начале количество деталей, которые требуются для производства детали с номером i, затем через пробел номера самих деталей. В списке нет повторяющихся номеров деталей. Сумма длин всех списков не превосходит 200 000.

Известно, что не существует циклических зависимостей в производстве деталей.

Формат выходных данных

В единственной строке выходного файла должно содержаться одно число: минимальное время (в секундах), необходимое для скорейшего производства детали с номером 1.

Примеры

details.in	details.out
3 100 200 300 1 2 0 2 2 1	300
4 2 3 4 5 2 3 2 1 3 0 2 1 3	9

Задача F. Эвакуация

Имя входного файла:	`evacuation.in`
Имя выходного файла:	`evacuation.out`
Ограничение по времени:	3 секунды
Ограничение по памяти:	64 мегабайта

Одна из Сверхсекретных организаций, чье название мы не имеем право разглашать, представляет собой сеть из N подземных бункеров, соединенных равными по длине туннелями, по которым из любого бункера можно добраться до любого другого (не обязательно напрямую). Связь с внешним миром осуществляется через специальные засекреченные выходы, которые расположены в некоторых из бункеров. Организации понадобилось составить план эвакуации персонала на случай экстренной ситуации. Для этого для каждого из бункеров необходимо узнать, сколько времени потребуется для того, чтобы добраться до ближайшего из выходов. Вам, как специалисту по таким задачам, поручено рассчитать необходимое время для каждого из бункеров по заданному описанию помещения Сверхсекретной организации. Для вашего же удобства бункеры занумерованы числами от 1 до N.

Формат входных данных

В первой строке записано число N, во второй — число K ($1 \leqslant N \leqslant 100\,000$, $1 \leqslant K \leqslant N$) — количество бункеров и количество выходов соответственно. Далее через пробел записаны K различных чисел от 1 до N, обозначающих номера бункеров, в которых расположены выходы. Потом идёт целое число M ($1 \leqslant M \leqslant 100\,000$) — количество туннелей. Далее вводятся M пар чисел — номера бункеров, соединенных туннелем. По каждому из туннелей можно двигаться в обе стороны. В организации не существует туннелей, ведущих из бункера в самого себя, зато может существовать более одного туннеля между парой бункеров.

Формат выходных данных

В первой строке выведите N чисел, разделённых пробелом — для каждого из бункеров минимальное время, необходимое чтобы добраться до выхода. Считайте, что время перемещения по одному туннелю равно 1. Во второй строке выведите N чисел — для каждого бункера номер ближайшего бункера с выходом, если таких несколько выведите бункер с наименьшим номером.

Примеры

evacuation.in	evacuation.out
3 1 2 3 1 2 3 1 2 3	1 0 1 2 2 2