

## Задача А. Мистер Флойд

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

Дан ориентированный взвешенный граф. Вам необходимо найти пару вершин, кратчайшее расстояние от одной из которых до другой максимально среди всех пар вершин.

### Формат входных данных

В первой строке вводится единственное число  $n$  ( $1 \leq n \leq 100$ ) – количество вершин графа. В следующих  $n$  строках по  $n$  чисел задается матрица смежности графа, где  $-1$  означает отсутствие ребра между вершинами, а любое неотрицательное число – присутствие ребра данного веса. На главной диагонали матрицы – всегда нули.

### Формат выходных данных

Выведите искомое максимальное кратчайшее расстояние.

### Пример

стандартный ввод	стандартный вывод
6 0 6 8 -1 -1 -1 5 0 5 -1 -1 -1 1 7 0 -1 -1 -1 -1 -1 -1 0 6 -1 -1 -1 -1 -1 0 3 -1 -1 -1 2 -1 0	9

## Задача В. Флойд - существование

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

Дан ориентированный взвешенный граф. По его матрице смежности нужно для каждой пары вершин определить, существует ли кратчайший путь между ними или нет.

### Формат входных данных

В первой строке входного файла записано единственное число  $n$  ( $1 \leq n \leq 100$ ) — количество вершин графа. В следующих  $n$  строках по  $n$  чисел — матрица смежности графа ( $j$ -е число в  $i$ -й строке соответствует весу ребра из вершины  $i$  в вершину  $j$ ): число 0 обозначает отсутствие ребра, а любое другое число — наличие ребра соответствующего веса. Все числа по модулю не превышают 100.

### Формат выходных данных

Выведите  $n$  строк по  $n$  чисел.  $j$ -е число в  $i$ -й строке должно соответствовать кратчайшему пути из вершины  $i$  в вершину  $j$ . Число должно быть равно 0, если пути не существует, 1, если существует кратчайший путь, и 2, если пути существуют, но бывают пути сколь угодно маленького веса.

### Пример

стандартный ввод	стандартный вывод
3	1 1 1
0 1 2	1 1 1
1 0 3	1 1 1
2 3 0	

### Замечание

Кратчайший путь может не существовать по двум причинам:

- нет ни одного пути
- есть пути сколь угодно маленького веса

## Задача С. Форд-Беллман

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

Дан ориентированный граф, в котором могут быть кратные ребра и петли. Каждое ребро имеет вес, выражающийся целым числом (возможно, отрицательным). Гарантируется, что циклы отрицательного веса отсутствуют.

Требуется посчитать длины кратчайших путей от вершины номер 1 до всех остальных вершин.

### Формат входных данных

Программа получает сначала число  $n$  ( $1 \leq n \leq 100$ ) — количество вершин графа и число  $m$  ( $0 \leq m \leq 10^4$ ) — количество ребер. В следующих строках идет  $m$  троек чисел, описывающих ребра: начало ребра, конец ребра и вес (вес — целое число от  $-100$  до  $100$ ).

### Формат выходных данных

Программа должна вывести  $n$  чисел — расстояния от вершины номер 1 до всех вершин графа. Если пути до соответствующей вершины не существует, вместо длины пути выведите число 30000.

### Пример

стандартный ввод	стандартный вывод
6 4 1 2 10 2 3 10 1 3 100 4 5 -10	0 10 20 30000 30000 30000

## Задача D. Цикл

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

Дан ориентированный граф. Определить, есть ли в нем цикл отрицательного веса, и если да, то вывести его.

### Формат входных данных

В первой строке содержится число  $n$  ( $1 \leq n \leq 100$ ) – количество вершин графа. В следующих  $n$  строках находится по  $n$  чисел – матрица смежности графа. Веса ребер по модулю меньше  $10^5$ . Если ребра нет, соответствующее значение равно  $10^5$ .

### Формат выходных данных

В первой строке выведите «YES», если цикл существует, или «NO» в противном случае. При наличии цикла выведите во второй строке количество вершин в нем (считая одинаковые – первую и последнюю), а в третьей строке – вершины, входящие в этот цикл, в порядке обхода. Если циклов несколько, то выведите любой из них.

### Пример

стандартный ввод	стандартный вывод
3	YES
100000 100000 -51	4
100 100000 100000	2 1 3 2
100000 -50 100000	

## Задача Е. Лабиринт знаний

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

В Летней Компьютерной Школе (ЛКШ) построили аттракцион «Лабиринт знаний». Лабиринт представляет собой  $n$  комнат, занумерованных от 1 до  $n$ , между некоторыми из которых есть двери. Когда человек проходит через дверь, показатель его знаний изменяется на определенную величину, фиксированную для данной двери. Вход в лабиринт находится в комнате 1, выход – в комнате  $n$ . Каждый ученик проходит лабиринт ровно один раз и попадает в ту или иную учебную группу в зависимости от количества набранных знаний (при входе в лабиринт этот показатель равен нулю). Ваша задача показать наилучший результат.

### Формат входных данных

Первая строка входных данных содержит целые числа  $n$  ( $1 \leq n \leq 2000$ ) – количество комнат и  $m$  ( $1 \leq m \leq 10000$ ) – количество дверей. В каждой из следующих  $m$  строк содержится описание двери – номера комнат, из которой она ведет и в которую она ведет (через дверь можно ходить только в одном направлении), а также целое число, которое прибавляется к количеству знаний при прохождении через дверь (это число по модулю не превышает 10000). Двери могут вести из комнаты в нее саму, между двумя комнатами может быть более одной двери.

### Формат выходных данных

Выведите «:» – если можно получить неограниченно большой запас знаний, «:(» – если лабиринт пройти нельзя, и максимальное количество набранных знаний в противном случае.

### Пример

стандартный ввод	стандартный вывод
2 2 1 2 3 1 2 7	7