

Задача А. Города и дороги

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

Формат входного файла

Во входном файле записано число N ($0 \leq N \leq 100$). В следующих N строках записано по N чисел, каждое из которых является единичкой или ноликом. Причем, если в позиции (i, j) квадратной матрицы стоит единичка, то i -й и j -й города соединены дорогами, а если нолик, то не соединены.

Формат выходного файла

В выходной файл выведите одно число — количество дорог в этой стране.

Примеры

<code>cities.in</code>	<code>cities.out</code>
5 0 1 0 0 0 1 0 1 1 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0	3

Задача В. Светофоры

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

В подземелье M тоннелей и N перекрестков, каждый тоннель соединяет какие-то два перекрестка. Мышиный король решил поставить по светофору в каждом тоннеле перед каждым перекрестком. Напишите программу, которая посчитает, сколько светофоров должно быть установлено на каждом из перекрестков. Перекрестки пронумерованы числами от 1 до N .

Формат входного файла

Во входном файле записано два числа N и M ($0 < N \leq 100$, $0 \leq M \leq \frac{N(N-1)}{2}$). В следующих M строках записаны по два числа i и j ($1 \leq i, j \leq N$), которые означают, что перекрестки i и j соединены тоннелем.

Формат выходного файла

В выходной файл вывести N чисел: k -е число означает количество светофоров на k -м перекрестке.

Примеры

<code>lights.in</code>	<code>lights.out</code>
7 10 5 1 3 2 7 1 5 2 7 4 6 5 6 4 7 5 2 1 5 3	3 3 2 2 5 2 3

Задача С. Цветной дождь

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

Формат входного файла

В первой строке файла записано число N — количество холмов ($1 \leq N \leq 100$). Во второй и далее — матрица смежности, описывающая наличие мостов между холмами. В последней строке написаны N чисел k_1, k_2, \dots, k_N , которые обозначают цвет соответствующего холма: 1 — красный, 2 — синий, 3 — зеленый.

Формат выходного файла

Выведите количество мостов, соединяющих холмы разных цветов.

Примеры

<code>rain.in</code>	<code>rain.out</code>
1 0 1	0
7 0 1 0 0 0 1 1 1 0 1 0 0 0 0 0 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1 0 1 0 1 0 1 0 0 1 0 0 0 0 0 0 1 1 1 1 1 3 3	4

Задача D. От матрицы смежности к списку ребер, неориентированный вариант

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

Простой неориентированный граф задан матрицей смежности, выведите его представление в виде списка ребер.

Формат входного файла

В первой строке файла — число N ($1 \leq N \leq 100$). Во второй и далее — матрица смежности.

Формат выходного файла

Выведите список ребер графа.

Примеры

<code>mtol.in</code>	<code>mtol.out</code>
3	1 2
0 1 1	1 3
1 0 1	2 3
1 1 0	

Задача E. От списка ребер к матрице смежности

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

Формат входного файла

В первой строке — числа N и M ($1 \leq N \leq 100$, $1 \leq M \leq \frac{N(N-1)}{2}$). Далее идет M строк — список ребер.

Формат выходного файла

Выведите матрицу смежности неориентированного графа.

Примеры

<code>ltom.in</code>	<code>ltom.out</code>
3 3	0 1 1
1 2	1 0 1
2 3	1 1 0
1 3	

Задача F. Транзитивность графа

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

Напомним, что граф называется транзитивным, если всегда из того, что вершины u

и v соединены ребром и вершины v и w соединены ребром следует, что вершины u и w соединены ребром.

Проверьте, что заданный неориентированный граф является транзитивным.

Формат входного файла

В первой строке — числа N и M ($1 \leq N \leq 100$, $1 \leq M \leq \frac{N(N-1)}{2}$). Далее идет M строк — список ребер.

Формат выходного файла

Выведите YES или NO, как ответ на вопрос о транзитивности графа.

Примеры

<code>transitive.in</code>	<code>transitive.out</code>
3 3 1 2 2 3 1 3	YES
3 2 1 2 1 3	NO

Задача G. Проверка на неориентированность

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

По матрице $N \times N$ из нулей и единиц определите, может ли данная матрица быть матрицей смежности простого неориентированного графа.

Формат входного файла

В первой строке число N ($1 \leq N \leq 1000$), далее матрица — N строк по N чисел, каждое из которых равно 0 или 1.

Формат выходного файла

Выведите YES, если приведенная матрица может быть матрицей смежности простого неориентированного графа, иначе выведите NO.

Примеры

check.in	check.out
3 0 1 1 1 0 1 1 1 0	YES
3 0 1 0 1 0 1 1 1 0	NO
3 0 1 0 1 1 1 0 1 0	YES

Задача Н. Подсчет количества ребер неориентированного графа

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

Простой неориентированный граф задан матрицей смежности. Найдите количество ребер в графе.

Формат входного файла

В первой строке число 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

Выведите количество ребер заданного графа.

Примеры

count.in	count.out
3 0 1 1 1 0 1 1 1 0	3

Задача I. Проверка на наличие параллельных ребер, ориентированный вариант

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

Ориентированный граф задан списком ребер. Проверьте, содержит ли он параллельные (кратные) ребра.

Формат входного файла

N — число вершин и M — число ребер ($1 \leq M \leq 10000$), затем M пар чисел — ребра графа.

Формат выходного файла

Выведите YES, если граф содержит параллельные ребра, иначе NO.

Примеры

check.in	check.out
5 3 2 5 3 1 3 2	NO

Задача J. Полустепени вершин

Имя входного файла: `half-degree.in`
Имя выходного файла: `half-degree.out`

Ориентированный граф задан матрицей смежности. Найдите полустепени захода и полустепени исхода всех вершин графа (т.е. количество входящих в нее и исходящих из нее ребер соответственно для каждой вершины).

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

Выведите N пар чисел — для каждой вершины сначала полустепень захода и затем полустепень исхода.

Примеры

half-degree.in	half-degree.out
4 0 1 0 1 1 0 1 1 0 1 0 0 1 1 1 1	2 2 3 3 2 1 3 4

Задача K. Истоки и стоки

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

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	2 3 4
0 0 0 0 0	3 1 4 5
0 0 0 0 1	
1 1 0 0 0	
0 0 0 0 0	
0 0 0 0 0	

Задача L. Полный граф

Имя входного файла: `complete.in`

Имя выходного файла: `complete.out`

Неориентированный граф называется полным, если любая пара его различных вершин соединена ребром. Для заданного списком ребер графа проверьте, является ли он полным.

Формат входного файла

N — число вершин ($1 \leq N \leq 100$) и M — число ребер, затем M пар чисел — ребра графа.

Формат выходного файла

Выведите YES, если граф является полным, и NO в противном случае.

Примеры

complete.in	complete.out
3 3	YES
1 2	
1 3	
2 3	