

Задача А. 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 нём выделена вершина. Вам необходимо найти количество вершин, лежащих с ней в одной компоненте связности (считая её саму).

Формат входного файла

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

Формат выходного файла

Выведите одно целое число — искомое количество вершин.

Примеры

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

Задача В. Компоненты связности

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

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

Формат входного файла

В первой строке входного файла содержится одно натуральное число N ($N \leq 100$) — 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 выходного файла

Вывести одно целое число — искомое количество компонент связности графа.

Примеры

<code>matrix.in</code>	<code>matrix.out</code>
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 С. Площадь комнаты

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

Требуется вычислить площадь комнаты в квадратном лабиринте.

Формат входного файла

В первой строке находится число N — размер лабиринта ($3 \leq N \leq 10$). В следующих N строках задан лабиринт (`.` — пустая клетка, `*` — стенка). Последняя строка содержит два числа — номер строки и столбца клетки, находящейся в комнате, площадь которой необходимо вычислить. Гарантируется, что эта клетка пустая и что лабиринт окружен стенками со всех сторон.

Формат выходного файла

Требуется вывести единственное число — количество пустых клеток в данной комнате.

Примеры

<code>area.in</code>	<code>area.out</code>
5 ***** **.* *.*.* *.*.* ***** 2 4	3