Задача A. Обход в ширину

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

Дан неориентированный граф. В нём необходимо найти расстояние от одной заданной вершины до другой.

Формат входных данных

В первой строке входного файла содержатся три натуральных числа N, S и F ($1 \leqslant S, F \leqslant N \leqslant 100$) — количество вершин в графе и номера начальной и конечной вершин соответственно. Далее в N строках задана матрица смежности графа. Если значение в j-м элементе i-й строки равно 1, то в графе есть направленное ребро из вершины i в вершину j.

Формат выходных данных

В единственной строке должно находиться минимальное расстояние от начальной вершины до конечной. Если пути не существует, выведите 0.

Примеры

bfs.in	bfs.out
5 4 2 0 1 0 1 1 1 0 1 0 0 0 1 0 1 0 1 0 1 0 0 1 0 0 0 0	2

Задача B. Путь конём

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

На шахматной доске 8×8 указаны две различные клетки. Найдите кратчайший маршрут коня из первой клетки во вторую.

Формат входных данных

Во входном файле записаны координаты двух клеток. Каждая координата представлена двумя символами, где сначала указана одна строчная буква от `a` до `h`, а после буквы (без пробела) цифра от `1` до `8`, например `h8`. Каждая клетка записана в отдельной строке. Гарантируется, что координаты клеток различны.

Формат выходных данных

Программа должна вывести последовательность клеток, первая из которых совпадает с первой данной, а последняя совпадает со второй данной. Две соседние клетки должны быть соединены ходом коня, при этом количество клеток в последовательности должно быть минимально возможным. Если существует несколько возможных ответов на задачу, разрешается выводить любой.

Примеры

knight.in	knight.out
a1 b1	a1 b3 d2 b1

Задача C. Дейкстра

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

Дан ориентированный взвешенный граф.
Найдите кратчайшее расстояние от одной заданной вершины до другой.

Формат входных данных

В первой строке входного файла три числа: N, S и F ($1 \leqslant N \leqslant 2000, 1 \leqslant S, F \leqslant N$), где N — количество вершин графа, S — начальная вершина, а F — конечная. В следующих N строках по N чисел — матрица смежности графа, где -1 означает отсутствие ребра между вершинами, а любое целое неотрицательное число, не превосходящее 10 000 — присутствие ребра данного веса. На главной диагонали матрицы всегда нули.

Формат выходных данных

Вывести искомое расстояние или -1, если пути не существует.

Примеры

dijkstra.in	dijkstra.out
3 1 2 0 -1 2 3 0 -1 -1 4 0	6

Задача D. Расстояние между вершинами

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

Дан неориентированный взвешенный граф.
Найти вес минимального пути между двумя вершинами.

Формат входных данных

Первая строка входного файла содержит натуральные числа N, M, вторая строка числа S и $F (N \leqslant 5\,000,\ M \leqslant 100\,000,\ 1 \leqslant S, F \leqslant N,\ S \neq F)$ — количество вершин и ребер графа а также номера вершин, длину пути между которыми требуется найти. Следующие M строк по три натуральных числа b_i, e_i и w_i — номера концов i-ого ребра и его вес соответственно $(1 \leqslant b_i, e_i \leqslant n,\ 0 \leqslant w_i \leqslant 100\,000)$.

Формат выходных данных

Первая строка должна содержать одно натуральное число — вес минимального пути между вершинами S и F. Во второй строке через пробел выведите вершины на кратчайшем пути из S в F в порядке обхода. Если путь из S в F не существует, выведите -1.

Примеры

distance.in	distance.out
4 4 1 3 1 2 1 2 3 2 3 4 5 4 1 4	3 1 2 3