

## Задача А. Обход в ширину

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

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

### Формат входного файла

В первой строке входного файла содержатся три натуральных числа  $N$ ,  $S$  и  $F$  ( $1 \leq S, F \leq N \leq 100$ ) — 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 выходного файла

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

### Примеры

bfs.in	bfs.out
5 5 3 0 0 1 1 0 0 0 0 1 0 1 0 0 0 1 1 1 0 0 1 0 0 1 1 0	1

## Задача В. Поиск цикла

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

Дан ориентированный невзвешенный граф. Необходимо определить есть ли в нём циклы, и если есть, то вывести любой из них.

### Формат входного файла

В первой строке входного файла находятся два натуральных числа  $N$  и  $M$  ( $1 \leq N \leq 100\,000$ ,  $M \leq 100\,000$ ) — количество вершин и рёбер в графе соответственно. Далее в  $M$  строках перечислены рёбра графа. Каждое ребро задаётся парой чисел — номерами начальной и конечной вершин соответственно.

### Формат выходного файла

Если в графе нет цикла, то вывести «NO», иначе — «YES» и затем перечислить все вершины в порядке обхода цикла.

### Примеры

cycle.in	cycle.out
2 2 1 2 2 1	YES 1 2
2 2 1 2 1 2	NO

## Задача С. Максимум по минимуму

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

Дан ориентированный невзвешенный граф. В нём необходимо найти вершину, кратчайшее расстояние от которой до заданной максимально.

### Формат входного файла

В первой строке входного файла содержатся три натуральных числа  $N$ ,  $M$  и  $S$  ( $1 \leq S \leq N \leq 5000$ ,  $1 \leq M \leq 20\,000$ ) — количество вершин и рёбер в графе и номер заданной вершины соответственно. Далее в  $M$  строках перечислены рёбра графа. Каждое ребро задаётся парой чисел — номерами начальной и конечной вершин соответственно.

### Формат выходного файла

Вывести одно целое число — искомое кратчайшее расстояние.

### Примеры

maxmin.in	maxmin.out
3 5 3 1 2 2 1 3 1 2 3 3 3	2

## Задача D. Выход из лабиринта

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

Во время торнадо в Костромской области было повалено большое количество деревьев. Однажды утром, спустя пару дней после торнадо, мальчик Илья вышел прогуляться и увидел, что поваленные деревья образовали настоящий лабиринт. Вася очень обрадовался,

стал гулять и играть в лабиринте. Внезапно ему позвонил преподаватель и поинтересовался, почему он давно не видел Илью. Чтобы не получить дырку в бэйджик за самовольный выход за территорию лагеря, Илья, конечно же, хотел попасть в домик как можно скорее, но и побродить по лабиринту ему тоже очень хотелось. Помогите ему узнать, как быстро он сможет добраться до домика.

### Формат входного файла

В первой строке входного файла заданы целые положительные числа  $N$  и  $M$  ( $1 \leq N, M \leq 1000$ ). В следующих  $N$  строках заданы по  $M$  символов, описывающих лабиринт. На позициях, по которым Илья может перемещаться, записан символ «0», а на позициях, которые перегорожены деревьями, записан символ «1». После описания лабиринта следуют целые числа  $x_1, y_1, x_2, y_2$  — координаты (то есть соответствующие номера столбцов и строк) Ильи и его домика соответственно ( $1 \leq x_1, x_2 \leq M, 1 \leq y_1, y_2 \leq N$ ).

### Формат выходного файла

В выходной файл выведите единственное число — длину кратчайшего пути от точки  $(x_1, y_1)$  до точки  $(x_2, y_2)$ , если Илья может добраться до domu по лабиринту, и «-1» в противном случае.

### Примеры

maze.in	maze.out
4 6 1 0 1 1 1 0 0 0 1 0 0 0 1 0 1 1 0 0 0 0 0 0 0 0 2 1 5 3	7
4 6 1 0 1 1 1 0 0 0 1 0 0 0 1 0 1 1 0 0 0 0 0 1 0 0 2 1 5 3	-1
5 5 0 0 0 0 0 0 1 1 1 0 0 1 1 1 0 0 1 1 1 0 0 0 0 0 0 1 3 1 3	0

### Задача E. TopSort

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

Дан ориентированный невзвешенный граф. Необходимо его топологически отсортировать.

### Формат входного файла

В первой строке входного файла два натуральных числа  $N$  и  $M$  ( $1 \leq N \leq 10^5, 1 \leq M \leq 10^5$ ) — количество вершин и рёбер в графе соответственно. Далее в  $M$  строках перечислены рёбра графа. Каждое ребро задается парой чисел — номерами начальной и конечной вершин соответственно.

### Формат выходного файла

Вывести любую топологическую сортировку графа в виде последовательности номеров вершин. Если граф невозможно топологически отсортировать, вывести -1.

### Примеры

topsort.in	topsort.out
6 6 1 2 3 2 4 2 2 5 6 5 4 6	4 6 3 1 2 5
3 3 1 2 2 3 3 1	-1