

## Задача А. Просто поток

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

Дана система из узлов и труб, по которым может течь вода. Для каждой трубы известна наибольшая скорость, с которой вода может протекать через нее. Известно, что вода течет по трубам таким образом, что за единицу времени в каждый узел (за исключением двух — источника и стока) втекает ровно столько воды, сколько из него вытекает.

Ваша задача — найти наибольшее количество воды, которое за единицу времени может протекать между источником и стоком, а также скорость течения воды по каждой из труб.

Трубы являются двусторонними, то есть вода в них может течь в любом направлении. Между любой парой узлов может быть более одной трубы.

### Формат входных данных

В первой строке записано натуральное число  $N$  — количество узлов в системе ( $2 \leq N \leq 100$ ). Известно, что источник имеет номер 1, а сток номер  $N$ . Во второй строке записано натуральное  $M$  ( $1 \leq M \leq 5000$ ) — количество труб в системе. Далее в  $M$  строках идет описание труб. Каждая труба задается тройкой целых чисел  $A_i, B_i, C_i$ , где  $A_i, B_i$  — номера узлов, которые соединяет данная труба ( $A_i \neq B_i$ ), а  $C_i$  ( $0 \leq C_i \leq 10^4$ ) — наибольшая допустимая скорость течения воды через данную трубу.

### Формат выходных данных

В первой строке выведите наибольшее количество воды, которое протекает между источником и стоком за единицу времени. Далее выведите  $M$  строк, в каждой из которых выведите скорость течения воды по соответствующей трубе. Если направление не совпадает с порядком узлов, заданным во входных данных, то выводите скорость со знаком минус. Числа выводите с точностью  $10^{-3}$ .

### Примеры

<code>flow.in</code>	<code>flow.out</code>
2	4
2	1
1 2 1	-3
2 1 3	

## Задача В. Химия!!!

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

Вася и Сережа играют в следующую игру. В некоторых клетках клетчатого листка Сережа рисует один из символов 'H', 'O', 'N' или 'C', после чего Вася должен провести между некоторыми находящимися в соседних клетках символами линии так, чтобы получилось корректное изображение химической молекулы. К сожалению, Сережа любит рисовать много символов, и Вася не может сразу определить, возможно ли вообще нарисовать линии нужным способом. Помогите ему написать программу, которая даст ответ на этот вопрос.

В этой задаче проведенные между символами химических элементов линии будем считать корректным изображением молекулы, если они удовлетворяют следующим условиям:

- каждая линия соединяет символы, нарисованные в соседних (по стороне) клетках,
- между каждой парой символов проведено не более одной линии,
- от каждого элемента отходит ровно столько линий, какова валентность этого элемента (1 для H, 2 для O, 3 для N, 4 для C),
- пустые клетки ни с чем не соединены, и
- хотя бы в одной клетке нарисован какой-то символ.

### Формат входных данных

Первая строка входного файла содержит два натуральных числа  $n$  и  $m$  ( $1 \leq n, m \leq 50$ ) — размеры листочка, на котором рисует Сережа. Далее следуют  $n$  строк по  $m$  символов в каждой, задающих конфигурацию химических элементов, которую нарисовал Сережа; пустые клетки задаются символом '.'.

### Формат выходных данных

В выходной файл выведите одно слово: 'Valid', если линии провести требуемым образом можно, и 'Invalid', если нельзя.

### Примеры

<code>molecule.in</code>	<code>molecule.out</code>
3 4 HOH. NSOH OO..	Valid
3 4 HOH. NSOH OONH	Invalid

## Задача С. Великая стена

Имя входного файла: wall.in  
Имя выходного файла: wall.out  
Ограничение по времени: 3 секунды  
Ограничение по памяти: 64 мегабайта

У короля Людовика двое сыновей. Они ненавидят друг друга, и король боится, что после его смерти страна будет уничтожена страшными войнами. Поэтому Людовик решил разделить свою страну на две части, в каждой из которых будет властвовать один из его сыновей. Он посадил их на трон в города  $A$  и  $B$ , и хочет построить минимально возможное количество фрагментов стены таким образом, чтобы не существовало пути из города  $A$  в город  $B$ .

Страну, в которой властвует Людовик, можно упрощенно представить в виде прямоугольника  $m \times n$ . В некоторых клетках этого прямоугольника расположены горы, по остальным же можно свободно перемещаться. Кроме этого, ландшафт в некоторых клетках удобен для строительства стены, в остальных же строительство невозможно.

При поездках по стране можно перемещаться из клетки в соседнюю по стороне, только если ни одна из этих клеток не содержит горы или построенного фрагмента стены.

### Формат входных данных

В первой строке входного файла содержатся числа  $m$  и  $n$  ( $1 \leq m, n \leq 50$ ). Во второй строке заданы числа  $k$  и  $l$ , где  $0 \leq k, l, k + l \leq mn - 2$ ,  $k$  — количество клеток, на которых расположены горы, а  $l$  — количество клеток, на которых можно строить стену. Естественно, что на горах строить стену нельзя. Следующие  $k$  строк содержат координаты клеток с горами  $x_i$  и  $y_i$ , а за ними следуют  $l$  строк, содержащие координаты клеток, на которых можно построить стену —  $x_j$  и  $y_j$ . Последние две строки содержат координаты городов  $A$  ( $x_A$  и  $y_A$ ) и  $B$  ( $x_B$  и  $y_B$ ) соответственно. Среди клеток, описанных в этих  $k + l + 2$  строках, нет двух совпадающих. Гарантируется, что  $1 \leq x_i, x_j, x_A, x_B \leq m$  и  $1 \leq y_i, y_j, y_A, y_B \leq n$ .

### Формат выходных данных

В первой строке выходного файла должно быть выведено минимальное количество фрагментов стены  $F$ , которые необходимо построить. В последующих  $F$  строках необходимо вывести один из возможных вариантов застройки.

Если невозможно произвести требуемую застройку, то необходимо вывести в выходной файл единственное число  $-1$ .

### Примеры

wall.in	wall.out
5 5	3
3 8	1 3
3 2	2 3
2 4	3 1
3 4	
3 1	
1 3	
2 3	
3 3	
4 3	
5 3	
1 4	
1 5	
2 1	
5 5	