

## Содержание

<b>Must have</b>	<b>2</b>
Задача 20А. Улиточки [0.1 (1.0), 256]	2
<b>Задачи здорового человека</b>	<b>3</b>
Задача 20В. Разрез [0.1 (0.7), 256]	3
Задача 20С. Декомпозиция потока [0.1 (0.7), 256]	4
Задача 20D. Molecule. Химия!!! [0.1 (0.7), 256]	5
<b>Для искателей острых ощущений</b>	<b>6</b>
Задача 20Е. В поисках невест [0.1 (0.7), 256]	6
Задача 20F. Великая стена [0.4 (2.0), 256]	7
Задача 20G. Perspective [0.1 (0.7), 256]	8

---

У вас не получается читать/выводить данные?  
Воспользуйтесь примерами (c++) (python).

Обратите внимание, входные данные лежат в **стандартном потоке ввода** (он же stdin), вывести ответ нужно в **стандартный поток вывода** (он же stdout).

## Must have

### Задача 20А. Улиточки [0.1 (1.0), 256]

Две улиточки Маша и Петя сейчас находятся в на лужайке с абрикосами и хотят добраться до своего домика. Лужайки пронумерованы числами от 1 до  $n$  и соединены дорожками (может быть несколько дорожек соединяющих две лужайки, могут быть дорожки, соединяющие лужайку с собой же). В виду соображений гигиены, если по дорожке проползла улиточка, то вторая по той же дорожке уже ползти не может. Помогите Пете и Маше добраться до домика.

#### Формат входных данных

В первой строке файла записаны четыре целых числа —  $n$ ,  $m$ ,  $s$  и  $t$  (количество лужаек, количество дорог, номер лужайки с абрикосами и номер домика). В следующих  $m$  строках записаны пары чисел. Пара чисел  $(x, y)$  означает, что есть дорожка с лужайки  $x$  до лужайки  $y$  (из-за особенностей улиток и местности дорожки односторонние). Ограничения:  $2 \leq n \leq 10^5, 0 \leq m \leq 10^5, s \neq t$ .

#### Формат выходных данных

Если существует решение, то выведите YES и на двух отдельных строчках сначала последовательность лужаек для Машеньки (дам нужно пропускать вперед), затем путь для Пети. Если решения не существует, выведите NO. Если решений несколько, выведите любое.

#### Пример

stdin	stdout
3 3 1 3	YES
1 2	1 3
1 3	1 2 3
2 3	

#### Замечание

Дан оргграф, найти два непересекающихся по ребрам пути из  $s$  в  $t$ , вывести вершины найденных путей.

## Задачи здорового человека

### Задача 20В. Разрез [0.1 (0.7), 256]

Дан неориентированный граф. Найдите минимальный разрез между вершинами 1 и  $n$ .

#### Формат входных данных

На первой строке входного файла содержится  $n$  ( $1 \leq n \leq 100$ ) — число вершин в графе и  $m$  ( $0 \leq m \leq 400$ ) — количество ребер. На следующих  $m$  строках входного файла содержится описание ребер. Ребро описывается номерами вершин, которые оно соединяет, и его пропускной способностью (положительное целое число, не превосходящее 10 000 000), при этом никакие две вершины не соединяются более чем одним ребром.

#### Формат выходных данных

На первой строке выходного файла должны содержаться количество ребер в минимальном разрезе и их суммарная пропускная способность. На следующей строке выведите возрастающую последовательность номеров ребер (ребра нумеруются в том порядке, в каком они были заданы во входном файле).

#### Пример

stdin	stdout
3 3	2 8
1 2 3	1 2
1 3 5	
3 2 7	

### Задача 20С. Декомпозиция потока [0.1 (0.7), 256]

Задан ориентированный граф, каждое ребро которого обладает целочисленной пропускной способностью. Найдите максимальный поток из вершины с номером 1 в вершину с номером  $n$  и постройте декомпозицию этого потока.

#### Формат входных данных

Первая строка входного файла содержит  $n$  и  $m$  — количество вершин и количество ребер графа ( $2 \leq n \leq 500$ ,  $1 \leq m \leq 10\,000$ ). Следующие  $m$  строк содержат по три числа: номера вершин, которые соединяет соответствующее ребро графа и его пропускную способность. Пропускные способности не превосходят  $10^9$ .

#### Формат выходных данных

В первую строку выходного файла выведите одно число — количество путей в декомпозиции максимального потока из вершины с номером 1 в вершину с номером  $n$ . Следующий строки должны содержать описания элементарных потоков, на который был разбит максимальный. Описание следует выводить в следующем формате: величина потока, количество ребер в пути, вдоль которого течет данный поток и номера ребер в этом пути. Ребра нужно выводить в порядке прохождения. Ребра нумеруются с единицы в порядке появления во входном файле.

#### Примеры

stdin	stdout
4 5	3
1 2 1	1 2 1 4
1 3 2	1 3 2 3 4
3 2 1	1 2 2 5
2 4 2	
3 4 1	

#### Замечание

Должно заходить масштабирование потока, толкающее по пути  $\min_e(c_e - f_e)$ .  
Будьте бдительны! Граф **ориентирован**.

### Задача 20D. Molecule. Химия!!! [0.1 (0.7), 256]

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.

#### Пример

stdin	stdout
3 4 HOH. NCOH OO..	Valid
3 4 HOH. NCOH OONH	Invalid

#### Замечание

Всё ещё про потоки.

## Для искателей острых ощущений

### Задача 20Е. В поисках невест [0.1 (0.7), 256]

Однажды король Флатландии решил отправить  $k$  своих сыновей на поиски невест. Всем известно, что во Флатландии  $n$  городов, некоторые из которых соединены дорогами. Король живет в столице, которая имеет номер 1, а город с номером  $n$  знаменит своими невестами.

Итак, король повелел, чтобы каждый из его сыновей добрался по дорогам из города 1 в город  $n$ . Поскольку, несмотря на обилие невест в городе  $n$ , красивых среди них не так много, сыновья опасаются друг друга. Поэтому они хотят добраться до цели таким образом, чтобы никакие два сына не проходили по одной и той же дороге (даже в разное время). Так как король любит своих сыновей, он хочет, чтобы среднее время сына в пути до города назначения было минимально.

#### Формат входных данных

В первой строке входного файла находятся числа  $n$ ,  $m$  и  $k$  — количество городов и дорог во Флатландии и сыновей короля, соответственно ( $2 \leq n \leq 200$ ,  $1 \leq m \leq 2000$ ,  $1 \leq k \leq 100$ ). Следующие  $m$  строк содержат по три целых положительных числа каждая — города, которые соединяет соответствующая дорога и время, которое требуется для ее прохождения (время не превышает  $10^6$ ). По дороге можно перемещаться в любом из двух направлений, два города могут быть соединены несколькими дорогами.

#### Формат выходных данных

Если выполнить повеление короля невозможно, выведите на первой строке число  $-1$ . В противном случае выведите на первой строке минимальное возможное среднее время (с точностью 5 знаков после десятичной точки), которое требуется сыновьям, чтобы добраться до города назначения, не менее чем с пятью знаками после десятичной точки. В следующих  $k$  строках выведите пути сыновей, сначала число дорог в пути, и затем номера дорог в пути в том порядке, в котором их следует проходить. Дороги нумеруются, начиная с единицы, в том порядке, в котором они заданы во входном файле.

#### Пример

stdin	stdout
5 8 2	3.00000
1 2 1	3 1 5 6
1 3 1	3 2 7 8
1 4 3	
2 5 5	
2 3 1	
3 5 1	
3 4 1	
5 4 1	

#### Замечание

Mincost потока.

**Задача 20F. Великая стена [0.4 (2.0), 256]**

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  $-1$ .

**Пример**

stdin	stdout
5 5	3
3 8	3 1
3 2	1 3
2 4	3 3
3 4	
3 1	
1 3	
2 3	
3 3	
4 3	
5 3	
1 4	
1 5	
2 1	
5 5	

### Задача 20G. Perspective [0.1 (0.7), 256]

Баскетбол. NBA. Несколько команд играют турнир. Команды разбиты на дивизионы. Есть игры внутри дивизиона и между командами из разных дивизионов. Ничей не бывают, в каждой игре кто-то выигрывает, кто-то проигрывает. У вас есть любимая команда. Вы знаете всё про дивизион, в котором эта команда играет. Какие-то игры уже сыграны. Про каждую команду дивизиона вам известно, сколько побед уже одержала эта команда и сколько игр ещё предстоит ей сыграть (включая и игры внутри дивизиона, и игры вне дивизиона). Про каждую пару команд дивизиона известно, сколько ещё игр им предстоит сыграть друг с другом. Определите, могут ли все оставшиеся игры быть сыграны так, чтобы в результате у вашей любимой команды было побед не меньше чем у любой другой в её дивизионе?

#### Формат входных данных

На первой строке число команд в дивизионе  $n$  ( $2 \leq n \leq 20$ ). Ваша любимая команда имеет номер 1. Следующая строка содержит  $n$  чисел,  $i$ -е число обозначает количество побед у  $i$ -й команды. Следующая строка содержит  $n$  чисел,  $i$ -е число обозначает количество предстоящих игр у  $i$ -й команды. Далее  $n$  строк содержат по  $n$  чисел,  $j$ -е число на  $i$ -й строке обозначает количество предстоящих матчей между  $i$ -й и  $j$ -й командой дивизиона. Гарантируется, что матрица симметрична, и на диагонали стоят нули. Все числа целые, неотрицательные, до 10 000.

#### Формат выходных данных

Выведите YES или NO.

#### Пример

stdin	stdout
3 1 2 2 1 1 1 0 0 0 0 0 0 0 0 0	YES
3 1 2 2 1 1 1 0 0 0 0 0 1 0 1 0	NO

#### Замечание

Что-то похожее было на практике.