

Задача А. Максимальный поток

Имя входного файла: `maxflow.in`
Имя выходного файла: `maxflow.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Задан ориентированный граф, каждое ребро которого обладает целочисленной пропускной способностью. Найдите максимальный поток из вершины с номером 1 в вершину с номером n .

Формат входных данных

Первая строка входного файла содержит n и m — количество вершин и количество ребер графа ($2 \leq n \leq 100$, $1 \leq m \leq 1000$). Следующие m строк содержат по три числа: номера вершин, которые соединяет соответствующее ребро графа и его пропускную способность. Пропускные способности не превосходят 10^5 .

Формат выходных данных

В выходной файл выведите одно число — величину максимального потока из вершины с номером 1 в вершину с номером n .

Примеры

<code>maxflow.in</code>	<code>maxflow.out</code>
4 5 1 2 1 1 3 2 3 2 1 2 4 2 3 4 1	3

Задача В. Максимальный поток

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

Вам задан ориентированный граф G . Каждое ребро имеет некоторую пропускную способность. Найдите максимальный поток между вершинами 1 и n .

Формат входных данных

Первая строка входного файла содержит n и m — число вершин и ребер в графе ($2 \leq n \leq 500$, $1 \leq m \leq 10\,000$). Последующие строки описывают ребра. Каждое ребро задается тремя числами: начальная вершина ребра, конечная вершина ребра и пропускная способность ребра. Пропускные способности не превосходят 10^9 .

Формат выходных данных

Выведите величину максимального потока между вершинами 1 и n .

Далее для каждого ребра выведите величину потока, текущую по этому ребру.

Примеры

стандартный ввод	стандартный вывод
4 5 1 2 1 1 3 2 3 2 1 2 4 2 3 4 1	3 1 2 1 2 1
4 5 1 2 2 1 3 2 2 3 1 2 4 2 3 4 2	4 2 2 0 2 2
4 3 1 2 1 1 3 1 1 4 4	4 0 0 4

Задача С. Разрез

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

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).

Примеры

стандартный ввод	стандартный вывод
6 8	2 6
1 2 3	1 2
1 3 3	
2 4 2	
2 5 2	
3 4 2	
3 5 2	
5 6 3	
4 6 3	

Задача D. В Тридевятиом Царстве, в Тридевятиом Государстве...

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Служба безопасности Тридевятиого Царства имеет обширную сеть секретных агентов в Стране Оз. Конечно, контрразведка Страны Оз хочет каким-то образом передавать ложные сведения этим агентам. Прямо сейчас, после долгой агентурной работы у них есть возможность завербовать любого агента Тридевятиого Царства и убедить их работать на Страну Оз.

Но каждый агент Тридевятиого Царства, работающий на Страну Оз, будет требовать значительных затрат из казны. Поэтому контрразведка Страны Оз хочет разработать такой план вербовки агентов, чтобы число завербованных агентов было минимально возможным.

Сеть агентов Тридевятиого Царства состоит из двух типов агентов: ведьм и леших. Ведьмы летают на метлах, исследуют Страну Оз и собирают о ней информацию, а лешие отправляют собранные сведения в Тридевятиое Царство с помощью лесных птиц. Списки леших, которые могут работать с каждой из ведьм, известны контрразведке Страны Оз.

Контрразведка Страны Оз также обладает информацией о текущем состоянии сети агентов Тридевятиого Царства. Каждая ведьма может иметь не больше одного *текущего лешего*. Другие лешие являются резервными для данной ведьмы и взаимодействие с ними будет производиться только в исключительных обстоятельствах. Никакие две ведьмы не могут иметь одного и того же текущего лешего. Кроме того, контрразведка Страны Оз обнаружила, что сеть агентов Тридевятиого Царства сейчас выстроена оптимальным образом согласно указанному правилу, то есть число ведьм, у которых есть текущий леший, максимально возможное.

Ваша задача — найти минимальное число агентов Тридевятиого Царства, которое нужно завербовать, и возможный план вербовки, чтобы информация, передаваемая каждой текущей или резервной парой ведьмы и лешего была фальсифицирована контрразведкой Страны Оз на стороне ведьмы или на стороне лешего.

Формат входных данных

Первая строка входного файла содержит числа m и n ($1 \leq m \leq 100\,000$, $1 \leq n \leq 100\,000$) и c ($0 \leq c \leq 500\,000$), где m — число ведьм, n — число леших и c — число связей между ними.

В следующих c строках перечислены пары чисел u_i и v_i ($0 \leq u_i \leq m - 1$, $0 \leq v_i \leq n - 1$), описывающих, что ведьма под номером u_i может взаимодействовать с лешим под номером v_i .

В последней строке входного файла содержится список текущих леших для ведьм. В ней перечислено m чисел $-1 \leq L_i \leq n - 1$ — номера текущих леших для каждой ведьмы, или -1 в случае если у очередной ведьмы нет текущего лешего.

Формат выходных данных

Первая строка выходного файла должна содержать минимальное возможное число агентов Тридевятиого Царства, которых нужно завербовать.

Вторая строка выходного файла должна содержать число ведьм S , которых надо завербовать, а также S номеров выбранных для вербовки ведьм в возрастающем порядке.

Третья строка выходного файла должна содержать описание завербованных леших в том же формате.

Примеры

стандартный ввод	стандартный вывод
3 2 4	2
1 1	1 0
0 1	1 1
2 1	
0 0	
0 1 -1	

Задача Е. Улиточки

Имя входного файла: `stdin`
Имя выходного файла: `stdout`
Ограничение по времени: 2.5 секунды
Ограничение по памяти: 256 мегабайт

Две улиточки Маша и Петя сейчас находятся 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). Ввиду 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 , a и h (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: $2 \leq n \leq 10^5$, $0 \leq m \leq 10^5$, $a \neq h$

Формат выходных данных

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

Примеры

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

Задача F. Великая стена

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
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

стандартный ввод	стандартный вывод
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	

Задача G. Чокнутый профессор

Имя входного файла:	stdin
Имя выходного файла:	stdout
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В Университете города М. проводят эксперимент. Преподаватели сами решают, что они будут читать в рамках того или иного курса. И вот преподаватель математического анализа (в простонародье — матана) оценил по некоторым критериям все известные ему темы в данном курсе. В результате этой ревизии каждой теме сопоставлено некоторое целое число (возможно, отрицательное) — полезность данной темы. Профессор хочет максимизировать суммарную полезность прочитанных им тем, но не все так просто. Для того что бы студенты поняли некоторые темы, необходимо, чтобы были прочитаны так же некоторые другие темы, так как некоторые доказательства базируются на фактах из других тем. Однако если существует цикл из зависимостей тем, то их все можно прочитать, и на качестве понимания материала студентами это не скажется.

Вас попросили составить список тем, которые профессор должен прочитать, таким образом, чтобы студенты все поняли, и суммарная полезность курса была максимальна.

Формат входных данных

Первая строка входного файла содержит одно число — N ($1 \leq N \leq 200$). Вторая строка содержит N целых чисел, не превосходящих по модулю 1000 — полезности каждой темы. Далее следуют N строк с описанием зависимостей тем. Каждое описание начинается количеством тем, которые необходимо понять для понимания данной темы. Потом следуют номера этих тем, разделенные пробелами.

Формат выходных данных

Выведите единственное число — максимально возможную суммарную полезность прочитанного материала.

Примеры

stdin	stdout
4 -1 1 -2 2 0 1 1 2 4 2 1 1	2
3 2 -1 -2 2 2 3 0 0	0

Задача Н. Транспортировка товаров

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

Вдоль трассы с односторонним движением расположены n городов. Города пронумерованы числами от 1 до n в порядке проезда вдоль трассы.

В i -м городе было произведено p_i единиц товара и может быть продано не более чем s_i единиц товара.

Для каждой пары городов i и j , таких что $1 \leq i < j \leq n$, можно **не более одного раза** перевезти **не более чем c** единиц товара из города i в город j . Заметьте, что товар можно перевозить только из города с меньшим номером в город с большим номером. **Перевозить товары между городами можно в любом порядке.**

Определите, какое максимальное количество произведённого товара суммарно может быть продано во всех городах после некоторой последовательности перевозок.

Формат входных данных

Первая строка входных данных содержит два числа n и c ($1 \leq n \leq 10\,000$, $0 \leq c \leq 10^9$) — количество городов и максимальное количество товаров, которое можно перевезти между городами за один раз.

Во второй строке записаны n чисел p_i ($0 \leq p_i \leq 10^9$) — количество единиц товара, произведённого в каждом городе.

В третьей строке записаны n чисел s_i ($0 \leq s_i \leq 10^9$) — количество единиц товара, которое может быть продано в каждом городе.

Формат выходных данных

Выведите одно число — максимальное количество товара, которое может быть продано во всех городах после некоторой последовательности перевозок.

Примеры

стандартный ввод	стандартный вывод
3 0 1 2 3 3 2 1	4
5 1 7 4 2 1 0 1 2 3 4 5	12
4 3 13 10 7 4 4 7 10 13	34

Замечание

Задача с финала Intel Code Challenge, прошедшего 08.10.2016 на сайте codeforces.com.

Задача I. В поисках невест

Имя входного файла: `brides.in`
Имя выходного файла: `brides.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

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.

Примеры

<code>brides.in</code>	<code>brides.out</code>
5 8 2	3.00000
1 2 1	2 2 6
1 3 1	2 3 8
1 4 3	
2 5 5	
2 3 1	
3 5 1	
3 4 1	
5 4 1	