

Задача А. Максимальный поток

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.25 секунд
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 — 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 .

Пример

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

Задача В. Разрез

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
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).

Пример

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 С. 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 .

Пример

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 D. Декомпозиция потока

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
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 и постройте декомпозицию этого потока.

Формат входных данных

Первая строка входного файла содержит n и m — количество вершин и количество ребер графа ($2 \leq n \leq 500$, $1 \leq m \leq 10000$). 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 нумеруются с единицы в порядке появления во входном файле.

Пример

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

Задача Е. Химия!!!

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

стандартный ввод	стандартный вывод
3 4 HOH. NSOH OO..	Valid
3 4 HOH. NSOH OONH	Invalid

Задача F. Нефтепроводы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Олигарх Вован, как и большинство других олигархов, занимается транспортировкой нефти из Западной Кукуляндии в Восточную Кукуляндию. В его владении находится огромная нефтедобывающая станция в Западной Кукуляндии, не меньшего размера нефтеперерабатывающая станция в Восточной Кукуляндии, а также система нефтепроводов, по которым нефть перегоняется из одной страны в другую. На столе у Вована лежит карта нефтепроводов. Хотелось бы знать, какое количество условных единиц нефти может транспортировать данная система. Каждый нефтепровод соединяет некоторую пару станций. На карте все станции пронумерованы, при этом добывающая станция имеет номер 1, перерабатывающая — номер N , а транзитные — номера от 2 до $N - 1$. Каждый нефтепровод может транспортировать ограниченное количество нефти, зато в любом направлении. Вован не знает, что Земля круглая, поэтому каждая станция на его карте имеет плоские координаты (x_i и y_i — координаты i -й станции). Нефтепроводы являются отрезками прямых. На карте пара нефтепроводов может пересекаться только по общей станции-вершине. Известно, что среди всех станций добывающая станция имеет наименьшую координату x , а перерабатывающая — наибольшую координату x .

Формат входных данных

В первой строке дано целое число N — количество станций на карте ($2 \leq N \leq 10000$). В следующих N строках перечислены координаты станций (x_i, y_i) через пробел. Координаты — целые числа, по модулю не превосходящие 10^8 . В следующей строке дано целое число M — количество нефтепроводов. Далее в M строках через пробел перечислены характеристики нефтепроводов — пара номеров станций, соединяемых нефтепроводом, а также пропускная способность нефтепровода в условных единицах — целое число от 1 до 10^8 . Гарантируется, что система нефтепроводов может транспортировать некоторое ненулевое количество нефти, но не может транспортировать более $2 \cdot 10^9$ условных единиц нефти.

Формат выходных данных

В первой строке выведите наибольшее количество условных единиц нефти, которое может транспортировать система Вована. В следующих M строках выведите план транспортировки — тройки чисел (A, B, C), означающие, что из станции A в станцию B должно течь C условных единиц нефти. Все нефтепроводы должны быть представлены в данном списке ровно один раз (даже те, по которым ничего не течёт). Число C во всех тройках должно быть неотрицательным.

Пример

стандартный ввод	стандартный вывод
3	1
0 0	1 2 1
1 1	2 3 1
2 0	
2	
1 2 2	
2 3 1	

Задача G. Разбиение на пары

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Космические археологи обнаружили на планете в соседней звездной системе n древних артефактов, которые они пронумеровали от 1 до n . Каждый артефакт имеет k различных параметров, каждый параметр характеризуется целым числом. Артефакт i имеет параметры $a_{i,1}, a_{i,2}, \dots, a_{i,k}$. Оказалось, что первые параметры у всех артефактов различны: для всех $i \neq j$ выполнено $a_{i,1} \neq a_{j,1}$, при этом другие параметры у артефактов могут совпадать.

Учёные также обнаружили текст, в соответствии с которым для активации артефактов их необходимо особым образом разбить на пары и совместить. Разбиение артефактов на пары является корректным, если для каждого t от 1 до k можно выбрать такое число b_t , что оно лежит на отрезке между значениями t -го параметра артефактов каждой пары. То есть, если артефакты i и j образуют пару, должно выполняться условие $a_{i,t} \leq b_t \leq a_{j,t}$ или условие $a_{i,t} \geq b_t \geq a_{j,t}$.

Теперь ученые хотят выяснить, верно ли расшифрован текст. Для этого необходимо проверить, существует ли корректное разбиение артефактов на пары. Каждый артефакт должен войти ровно в одну пару в разбиении.

Требуется написать программу, которая по описанию параметров артефактов определяет, можно ли разбить их на пары таким образом, чтобы для каждого параметра существовало значение, лежащее между значениями этого параметра артефактов каждой пары, и в случае положительного ответа выводит такое разбиение.

Формат входных данных

В первой строке заданы целые числа n и k — количество артефактов и количество параметров ($2 \leq n \leq 2 \cdot 10^5$, n чётно, $1 \leq k \leq 7$).

В следующих n строках задано по k целых чисел $a_{i,1}, a_{i,2}, \dots, a_{i,k}$ — параметры артефактов ($-10^9 \leq a_{i,j} \leq 10^9$, все значения $a_{i,1}$ различны).

Формат выходных данных

Выведите «NO», если требуемого разбиения на пары не существует.

В противном случае выведите «YES» в первой строке. Далее выведите $n/2$ строк, в каждой строке выведите по два числа — номера артефактов, из которых следует составить пару. Каждый артефакт должен быть выведен ровно один раз. Если существует несколько корректных разбиений артефактов на пары, разрешается вывести любое из них.

Примеры

стандартный ввод	стандартный вывод
6 2 8 6 1 5 6 3 3 1 4 7 7 2	YES 4 1 5 6 2 3
4 3 1 -1 -1 2 1 1 3 -1 1 4 1 -1	NO