

## Задача А. Максимальный поток на минималках

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

Костя поспорил с Ваней, что сможет написать самый примитивный поиск максимального потока за 15 минут. К сожалению, Костя прослушал лекцию о поиске максимального потока, поэтому теперь ему нужна ваша помощь.

### Формат входных данных

В первой строке входного файла содержится два числа:  $n$  и  $m$  ( $2 \leq n \leq 10$ ,  $1 \leq m \leq n \cdot (n - 1)$ ). Это количество вершин и рёбер в графе, в котором вам требуется найти поток. Далее следуют описания рёбер графа, по одному в каждой строке входного файла. Описание ребра состоит из трёх чисел:  $a$ ,  $b$ ,  $c$  ( $1 \leq a, b \leq n$ ,  $a \neq b$ ,  $1 \leq c \leq 100$ ). Эти числа означают, что из вершины  $a$  в вершину  $b$  идёт ребро пропускной способности  $c$ . Гарантируется, что в графе нет кратных рёбер.

### Формат выходных данных

В единственную строку выходного файла выведите одно число — размер максимального потока из вершины 1 в вершину  $n$ .

### Пример

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

## Задача В. 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 C. 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 D. Улиточки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 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.

### Пример

стандартный ввод	стандартный вывод
3 3 1 3	YES
1 2	1 3
1 3	1 2 3
2 3	

## Задача E. 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 F. 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
<pre>3 4 НОН. НСОН ОО..</pre>	Valid
<pre>3 4 НОН. НСОН ООNH</pre>	Invalid

## Задача G. Живопись

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

В стране Олимпия очень развита живопись. Картиной считается любой прямоугольник, который состоит из черных и белых единичных квадратов. Художник Олимпус решил радикально улучшить свои картины. Для этого он планирует к белому и черному цветам добавить еще и серый оттенок. По его задумке, граница между каждым черным и белым квадратом должна содержать серую линию, чтобы образовался эффект плавного перехода.

Однако, перед началом работы, он обнаружил, что серая краска очень дорого стоит. Чтобы сэкономить деньги художник решил оценить, не выгоднее ли сначала перекрасить некоторые белые квадраты в черные, а черные в белые для того, чтобы минимизировать расходы на краску.

Напишите программу, которая по информации о существующей картине определяет минимальную сумму денег, которые понадобятся на ее улучшение.

### Формат входных данных

Первая строка входного файла содержит пять натуральных чисел  $N$ ,  $M$ ,  $w$ ,  $b$ ,  $g$ . ( $1 \leq N, M \leq 70$ ) — высота и ширина картины, ( $1 \leq w, b, g \leq 1000$ ) — цена рисования одного белого единичного квадрата, черного единичного квадрата и серой линии единичной длины, соответственно.

Далее следует  $N$  строк, каждая из которых состоит из  $M$  литер. Литера B соответствует черному квадрату, а W — белому.

### Формат выходных данных

Единственная строка выходного файла должна содержать одно целое число, которое есть минимальной суммой затрат на улучшение картины.

### Пример

стандартный ввод	стандартный вывод
3 2 10 12 1 BW WB BW	7

## Задача N. Чаепитие

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

В одном из отделов крупной организации работает  $n$  человек. Как практически все сотрудники этой организации, они любят пить чай в перерывах между работой. При этом они достаточно дисциплинированы и делают в день ровно один перерыв, во время которого пьют чай. Для того, чтобы этот перерыв был максимально приятным, каждый из сотрудников этого отдела обязательно пьет чай одного из своих любимых сортов. В разные дни сотрудник может пить чай разных сортов. Для удобства пронумеруем сорта чая числами от 1 до  $m$ .

Недавно сотрудники отдела купили себе большой набор чайных пакетиков, который содержит  $a_1$  пакетиков чая сорта номер 1,  $a_2$  пакетиков чая сорта номер 2, ...,  $a_m$  пакетиков чая сорта номер  $m$ . Теперь они хотят знать, на какое максимальное число дней им может хватить купленного набора так, чтобы в каждый из дней каждому из сотрудников доставался пакетик чая одного из его любимых сортов.

Каждый сотрудник отдела пьет в день ровно одну чашку чая, которую заваривает из одного пакетика. При этом пакетики чая не завариваются повторно.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 50$ ). Вторая строка содержит  $m$  целых чисел  $a_1, \dots, a_m$  ( $1 \leq a_i \leq 10^6$  для всех  $i$  от 1 до  $m$ ).

Далее следуют  $n$  строк —  $i$ -я из этих строк описывает любимые сорта  $i$ -го сотрудника отдела и имеет следующий формат: сначала следует положительное число  $k_i$  — количество любимых сортов чая этого сотрудника, а затем идут  $k_i$  различных чисел от 1 до  $m$  — номера этих сортов.

### Формат выходных данных

Выведите одно целое число — искомое максимальное количество дней.

### Пример

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

## Задача I. 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

## Задача J. Чокнутый профессор

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

стандартный ввод	стандартный вывод
<pre>4 -1 1 -2 2 0 1 1 2 4 2 1 1</pre>	<pre>2</pre>
<pre>3 2 -1 -2 2 2 3 0 0</pre>	<pre>0</pre>

## Задача К. Две перестановки

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

Даны две перестановки  $p_0, p_1, \dots, p_{n-1}$  и  $q_0, q_1, \dots, q_{n-1}$  чисел  $0, 1, \dots, n-1$ . Надо построить две перестановки  $A$  и  $B$  чисел  $0, 1, \dots, n-1$ , удовлетворяющие следующим условиям:

- Для каждого  $i$  ( $0 \leq i < n$ )  $A_i$  должно быть равно  $i$  или  $p_i$ .
- Для каждого  $i$  ( $0 \leq i < n$ )  $B_i$  должно быть равно  $i$  или  $q_i$ .

Расстоянием между перестановками  $A$  и  $B$  назовём количество индексов  $i$ , что  $A_i \neq B_i$ . Найдите максимальное возможное расстояние между  $A$  и  $B$ .

### Формат входных данных

Первая строка содержит единственное число  $n$  ( $1 \leq n \leq 10^5$ ) — размер перестановок.

Вторая строка содержит  $n$  чисел  $p_0, p_1, \dots, p_{n-1}$  ( $0 \leq p_i < n$ ) — первая перестановка.

Третья строка содержит  $n$  чисел  $q_0, q_1, \dots, q_{n-1}$  ( $0 \leq q_i < n$ ) — вторая перестановка.

### Формат выходных данных

Выведите максимальное возможное расстояние между  $A$  и  $B$ .

### Примеры

стандартный ввод	стандартный вывод
4 2 1 3 0 0 2 3 1	3
10 0 4 5 3 7 8 2 1 9 6 3 8 5 6 4 0 2 1 7 9	8

## Задача L. Мадокa и первая сессия

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

О нет, на первой же сессии Мадокe попался билет со следующей сложной задачей:

Дано число  $n$  и  $m$  пар чисел  $(v_i, u_i)$ . А также есть массив  $b_1, b_2, \dots, b_n$ , **изначально заполненный нулями**.

Затем для каждого индекса  $i$ , где  $1 \leq i \leq m$ , выполняется либо  $b_{v_i} := b_{v_i} - 1$  и  $b_{u_i} := b_{u_i} + 1$ , либо  $b_{v_i} := b_{v_i} + 1$  и  $b_{u_i} := b_{u_i} - 1$ . Обратите внимание, что ровно одна из этих операций должна быть выполнена для каждого  $i$ .

Также дан массив  $s$  размера  $n$ , состоящий только из 0 и 1. И массив  $a_1, a_2, \dots, a_n$ , где гарантируется, что если  $s_i = 0$ , то  $a_i = 0$ .

Помогите Мадокe и определите, можно ли выполнить операции выше таким образом, чтобы для каждого  $i$ , где  $s_i = 1$ , выполнялось  $a_i = b_i$ . И если возможно, то как это сделать.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $m$  ( $2 \leq n \leq 100000, 1 \leq m \leq 100000$ ) — длина массива  $a$  и количество пар чисел.

Вторая строка содержит  $n$  целых чисел  $s_1, s_2, \dots, s_n$  ( $0 \leq s_i \leq 1$ ) — элементы массива  $s$ .

Третья строка содержит  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $|a_i| \leq m$ ) — элементы массива  $a$ . Гарантируется, что если  $s_i = 0$ , то  $a_i = 0$ .

$i$ -я из следующих  $m$  строк содержат два целых числа  $v_i$  и  $u_i$  ( $1 \leq v_i, u_i \leq n, v_i \neq u_i$ ) — индексы элементов массива  $b$ , к которым применяется операция. Также гарантируется, что не существует таких двух индексов  $i$  и  $j$ , где  $1 \leq i < j \leq m$ , что  $(v_i, u_i) = (v_j, u_j)$  или  $(v_i, u_i) = (u_j, v_j)$ .

### Формат выходных данных

Выведите в первой строке «YES», если можно выполнить операции нужным образом, и «NO» в противном случае.

Вы можете выводить каждую букву в любом регистре (например, «YES», «Yes», «yes», «yEs» будут распознаны как положительный ответ).

В случае, если вы вывели «YES», выведите  $m$  пар целых чисел. Если для пары  $(v_i, u_i)$  нужно выполнить  $b_{v_i} := b_{v_i} - 1$  и  $b_{u_i} := b_{u_i} + 1$ , выведите  $(v_i, u_i)$ . Иначе выведите  $(u_i, v_i)$ . Если существует несколько способов получить правильный ответ, можно вывести любой из них.

Пары можно выводить в любом порядке.

## Примеры

стандартный ввод	стандартный вывод
5 5 1 1 1 1 1 -2 0 2 1 -1 1 5 1 4 3 5 3 4 4 5	YES 1 5 1 4 5 3 4 3 5 4
5 5 0 1 0 1 0 0 1 0 0 0 1 3 2 3 3 5 3 4 4 5	YES 3 1 3 2 5 3 3 4 4 5
4 4 1 1 1 1 0 2 -2 2 1 3 1 4 2 3 2 4	NO

## Замечание

В первом примере массив  $b$  будет меняться следующим образом:  
 $[0, 0, 0, 0, 0] \rightarrow [-1, 0, 0, 1, 0] \rightarrow [-2, 0, 0, 1, 1] \rightarrow [-2, 0, 1, 0, 1] \rightarrow [-2, 0, 2, 0, 0] \rightarrow [-2, 0, 2, 1, -1]$ .  
 $a_i = b_i$  для всех индексов  $i$  от 1 до 5.

Во втором примере нам достаточно, чтобы в конце  $b_2 = 1$ , поскольку только  $s_2 = 1$ .

В третьем примере входных данных нельзя выполнить операции нужным образом.