

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. 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 С. 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. NCOH OO..	Valid
3 4 HOH. NCOH OONH	Invalid

Задача D. 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 Е. 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 1 2 3 1 3 3 2 4 2 2 5 2 3 4 2 3 5 2 5 6 3 4 6 3	2 6 1 2

Задача F. 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 G. 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 Н. 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 I. 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
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 J. 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

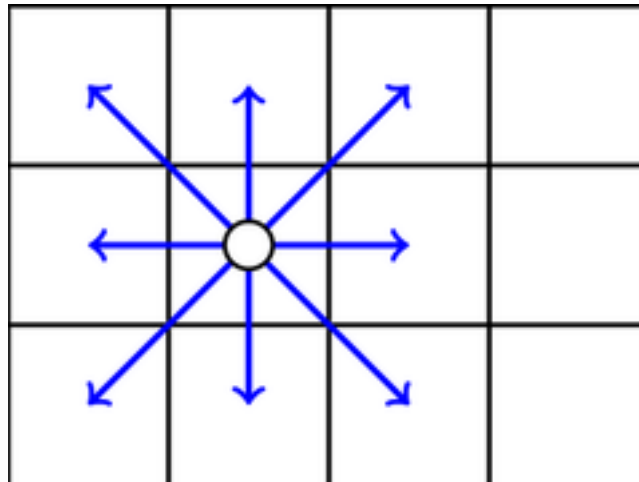
Задача К. Полезные ископаемые

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

Ведется проект по освоению планеты соседней звездной системы. Для добычи полезных ископаемых планируется направить на планету несколько партий роботов.

Участок поверхности планеты, на котором планируется добывать полезные ископаемые, представляет собой клетчатый прямоугольник размером w на h , клетки участка имеют координаты от $(1, 1)$ до (w, h) . В некоторых клетках участка находятся базы специалистов, в которые могут быть доставлены партии роботов. Всего на участке размещено s баз, и i -я база находится в клетке с координатами (x_i, y_i) .

Каждая партия роботов характеризуется тремя параметрами: j -я партия доставляется на базу b_j , содержит n_j роботов и каждый робот партии обладает мобильностью m_j . Когда партия роботов доставляется на соответствующую базу, каждый робот этой партии перемещается по поверхности планеты от базы до некоторой клетки. Если мобильность робота равна m , он может не более m раз переместиться на одну из восьми соседних клеток, как показано на рис. 1.



После того как роботы из всех доставленных партий размещаются на участке, они активируются и начинают добычу полезных ископаемых. В процессе перемещения в одной клетке может одновременно находиться произвольное количество роботов. Однако после активации в каждой клетке должно находиться не более q роботов.

Руководством проекта получена информация о t партиях роботов, которые могут быть последовательно отправлены на планету. После доставки всех партий роботов, учитывая их ограниченную мобильность, возможна ситуация, что не удастся разместить роботов на участке так, чтобы в каждой клетке оказалось не больше q роботов. Поэтому руководство должно выбрать k первых партий роботов, где $0 \leq k \leq t$, которые будут полностью доставлены на соответствующие базы. После этого, если $k < t$, следует дополнительно принять z из n_{k+1} роботов следующей, $(k + 1)$ -й партии, $0 \leq z < n_{k+1}$.

Все полученные таким образом роботы должны с учетом ограничения на мобильность разместиться на участке таким образом, чтобы в каждой клетке было не более q роботов. После этого они будут активированы и начнут добычу полезных ископаемых. Разумеется, руководство проекта старается максимизировать количество роботов, которые будут доставлены на планету, поэтому, с учетом описанных ограничений, требуется максимизировать k , а затем максимизировать z .

Требуется написать программу, которая по размерам участка, числу q , описанию расположения баз, а также количеству запланированных партий роботов и их описанию определяет максимальное число k — количество партий роботов, и затем — максимальное число z — дополнительное количество роботов из $(k + 1)$ -й партии, чтобы, доставив роботов на планету, их можно было разместить на участке таким образом, чтобы в каждой клетке оказалось не более q роботов.

Формат входных данных

Первая строка входного файла содержит числа w , h , s и q ($1 \leq w, h \leq 10^5$, $1 \leq s \leq 4$, $1 \leq q \leq 100$). Последующие s строк содержат по два целых числа x_i , y_i и описывают базы специалистов ($1 \leq x_i \leq w$, $1 \leq y_i \leq h$).

Следующая строка содержит число t – количество партий роботов ($1 \leq t \leq 100$). Последующие t строк описывают партии роботов и содержат по 3 целых числа: b_j , n_j и m_j ($1 \leq b_j \leq s$, $1 \leq n_j \leq w \cdot h \cdot q$, $0 \leq m_j < \max(w, h)$).

Формат выходных данных

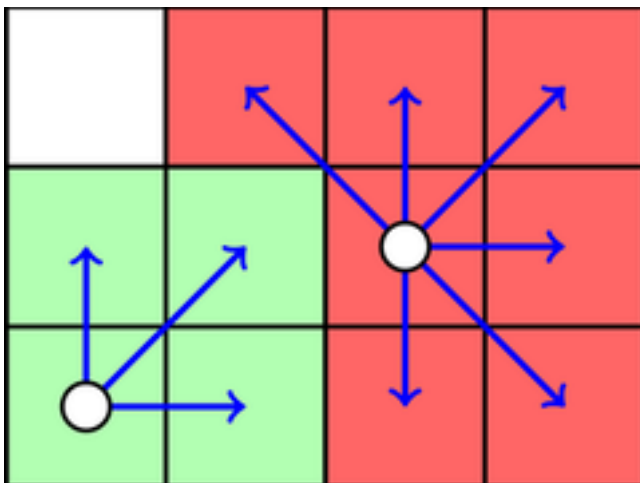
Требуется вывести два числа: k и z , $0 \leq k \leq t$. Если $k = t$, то z должно быть равно 0, иначе должно выполняться условие $0 \leq z < n_{k+1}$.

Пример

стандартный ввод	стандартный вывод
4 3 2 1 1 1 3 2 3 1 4 1 2 9 1 1 12 2	1 7

Замечание

В приведенном примере описания входных данных следует полностью принять первую партию роботов и дополнительно принять 7 роботов из второй партии. На рис. 2 показано, как можно разместить этих роботов на участке, чтобы в каждой клетке было не более одного робота. Базы специалистов показаны кружками. Клетки, в которых окажутся роботы с базы 1, показаны зелёным, а клетки, в которых окажутся роботы с базы 2, показаны красным цветом.



Возможное размещение роботов на участке в данном примере.

Задача L. Студенты в лесу

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

Однажды Иван Смирнов пообещал всех студентов, которые не ходят на пары, увезти с завязанными глазами в лес. И вот он выполнил свое обещание и вывез всех прогульщиков в лес, который представляет клетчатое поле $n \times m$. Вам выдана карта этого леса, клетки, занятые деревьями, обозначены символом «#», а свободные клетки — символом «.».

К счастью студентов их преподаватель не был извергом, и в этом лесу вполне себе можно было выживать. Более того, в лесу был бесплатный Wi-Fi, поэтому все студенты решили написать раунд на `codeforces`. Все студенты ВШЭ очень честные, а поэтому не могут писать раунд, находясь в соседних клетках.

Вам требуется рассадить наибольшее количество студентов в лесу так, чтобы никакие два студента не находились в соседних клетках.

Формат входных данных

В первой строке задана пара чисел n и m ($1 \leq n, m \leq 100$).

В следующих n строках содержатся строки, по m символов каждая, описывающие лес.

Формат выходных данных

Выведите поле $n \times m$ с максимальным количеством студентов. Студенты обозначаются символом «A».

Примеры

стандартный ввод	стандартный вывод
3 6 #.##.# #.##.#	#A##A# A.A..A #A##A#
3 3	A.A .A. A.A

Задача М. Yesterday

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

Мало кто знает, но Пол Маккартни помимо сочинения музыки увлекался нумерологией. Больше всего Полу нравились числа от 0 до $2^n - 1$, которые при записи в виде n -битных чисел содержали одинаковое количество нулей и единиц. Пусть $x_0, \dots, x_{m-1}$ — это такие числа, записанные в возрастающем порядке.

На одно из дней рождений Полу подарили перестановку $\pi_0, \dots, \pi_{m-1}$ ($0 \leq \pi_i < m$, все π_i различны) со следующим замечательным свойством: для любого i числа x_i и x_{π_i} различаются в двоичной записи ровно в k битах. Пол очень любил эту перестановку, носил ее с собой на репетиции, гулял с ней и бла-бла-бла.

Потом Пол потерял перестановку, сильно расстроился и написал песню Yesterday. В то время не было ни производительных ЭВМ, и было придумано не так уж и много алгоритмов, так что новую перестановку с такими же свойствами найти никто не смог.

Но сейчас все это есть, так что у вас нет никаких оправданий, чтобы не найти такую перестановку!

Формат входных данных

В единственной строке указана пара чисел n и k ($2 \leq n \leq 20$, $0 \leq k \leq n$, n — четное).

Формат выходных данных

Выведите подходящую перестановку, если такая существует, или -1 , в ином случае.

Примеры

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

Замечание

При $n = 4$ массив x выглядит следующим образом: $x = [3, 5, 6, 9, 10, 12]$.

Задача N. Спонсоры

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

Чернорусы решили провести первую в мире чернорусскую олимпиаду (ЧРО). Чтобы провести такое масштабное мероприятие, они прибегли к помощи n спонсоров. У спонсоров есть требования к названиям задач. Название каждой задачи олимпиады — это некоторая строка, состоящая из m строчных букв латинского алфавита. i -й спонсор требует, чтобы в названии одной из задач на позициях l_i до r_i встречалась подстрока w_i . Иными словами, в наборе задач должна быть задача с названием s , где $s_{l_i} \dots s_{r_i} = w_i$.

Конечно, можно просто составить набор из n задач, чтобы угодить каждому из спонсоров, но составлять новые задачи сложно, а поэтому требуется составить минимальный набор из k названий задач так, чтобы удовлетворить требованиям всех спонсоров.

Формат входных данных

В первой строке содержится пара чисел n и m ($1 \leq n \leq 500$, $1 \leq m \leq 1000$).

В каждой из последующих n строк содержится по паре чисел l_i , r_i и строка w_i , состоящая из строчных букв латинского алфавита ($1 \leq l_i \leq r_i \leq m$, $|w_i| = r_i - l_i + 1$) — требование на название, которое предлагает i -й спонсор.

Формат выходных данных

В первой строке выведите число k ($1 \leq k \leq n$) — минимальное количество задач, которое нужно составить чернорусам.

В каждой из k последующих строк выведите названия задач — строки из строчных латинских букв длиной m .

Обратите внимание, что выведенные строки должны удовлетворять условию задачи, то есть для каждого спонсора должно найтись хотя бы одно название, которое удовлетворяет его требованиям.

Примеры

стандартный ввод	стандартный вывод
5 7 3 3 a 5 7 aba 3 5 aca 1 5 abaca 1 4 aaaa	2 abacaba aaaaaba
8 18 6 10 brest 11 17 vitebsk 1 5 gomel 6 11 grodno 1 5 minsk 12 18 mogilev 3 3 n 10 10 w	3 gomelbrestvitebska minskgrodnogilev aaaaaaaaaaaaaaaa
5 9 1 9 aaaaaaaaaa 2 8 aaaaaaaa 3 7 aaaaaa 4 6 aaa 5 5 a	1 aaaaaaaaaa

Задача О. Регулярная задача

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2.5 секунд
Ограничение по памяти: 256 мегабайт

Вам дан граф из n вершин и $2n$ ребер такой, что степень каждой вершины равна 4. Требуется выделить в нем не пересекающиеся по вершинам циклы суммарной длиной n .

Гарантируется, что для каждого набора входных данных это возможно.

Формат входных данных

В первой строке указано число n ($1 \leq n \leq 500\,000$).

В следующих $2n$ строках указаны пары чисел u и v ($1 \leq u, v \leq n, u \neq v$) — ребра графа. Гарантируется, что в графе нет кратных ребер.

Формат выходных данных

В первой строке укажите количество циклов.

В каждой последующей строке следует описать очередной цикл. Сначала указывается длина цикла, а затем его вершины, в порядке обхода.

Пример

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

Задача Р. Гусеницы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	1024 мегабайта

На числовой прямой выстроилось множество гусениц. Известно, что гусеницы разбиты на группы, при этом в i -й группе c_i гусениц, а находится эта группа в позиции x_i .

Есть группы гусениц, которые любят ходить к другим гусеницам в гости, а есть гусеницы, которые принимают гостей. Именно, тип групп гусениц задан параметром t , и если $t_i = 1$, то гусеницы из i -й группы ходят в гости (то есть каждая из этой группы), а если $t_i = 2$, то гусеницы из i -й группы принимают гостей.

У каждой гусеницы также есть уровень зрения, который совпадает для всех гусениц в одной группе и для группы i равен v_i . Если суммарный уровень зрения некоторых двух гусениц, одна из которых в группе i с $t_i = 1$, а другая – в группе j с $t_j = 2$, превосходит расстояние между этими гусеницами, то гусенице из группы i неинтересно идти к гусенице из группы j в гости, ведь они хотят за чашкой чая узнавать что-то новое о числовой прямой, и им интересно обсуждать те её участки, которые не видит ни одна из пары этих гусениц (если одна видит какой-то участок, а другая нет, то одна может позвонить другой по телефону и рассказать это). Наоборот, если суммарный уровень зрения этих гусениц не больше расстояния между ними, то этим двум гусеницам будет интересно обсудить пройденный гусеницей из группы i путь за чашкой чая.

Более формально, гусеница из группы i может пойти в гости к гусенице из группы j только в том случае, если $|x_j - x_i| \geq v_j + v_i$ и $t_i = 1$ и $t_j = 2$.

Гусеницы из одной ходящей в гости группы в итоге могут разойтись по разным группам, также как и гусеницы одной принимающей гостей группы могут принимать гусениц из разных групп. В итоге каждая гусеница из путешествующей группы хочет прийти к какой-то в гости, так как любит путешествовать и не может сидеть дома. При этом принимающие гости гусеницы любят гостей, но вполне себе смиряются, если к ним никто не пройдет. Итак, ваша задача – по имеющимся данным о координатах, типах, количествах и уровнях зрения групп гусениц определить, возможно ли распределить всех гусениц первого типа по гусеницам второго типа, чтобы выполнялось описанное выше ограничение.

Также обратите внимание, что может возникнуть ситуации, когда есть гусеницы только первого типа (тогда их невозможно распределить) или только второго типа (тогда ответ «Yes»).

Формат входных данных

В первой строке вам дано число $1 \leq t \leq 3 \cdot 10^5$, обозначающее количество тестовых примеров.

Следующие строки содержат описания тестовых примеров. Каждый тестовый пример содержит следующую информацию. В первой строке вам дано единственное число – количество групп гусениц $1 \leq n \leq 3 \cdot 10^5$. В следующей строке вам даны координаты групп гусениц в порядке возрастания – n целых чисел $-10^9 \leq x_1 < x_2 < x_3 < \dots < x_n \leq 10^9$. В следующей строке вам даны количества гусениц в порядке слева направо – n натуральных чисел $1 \leq c_i \leq 10^9$. В следующей строке вам даны n параметров $1 \leq v_i \leq 10^9$ – натуральные числа, обозначающие уровни зрения. В следующей строке вам даны типы групп гусениц слева направо – n натуральных чисел $1 \leq t_i \leq 2$. Гарантируется, что сумма n по всем тестовым наборам не превосходит $3 \cdot 10^5$.

Формат выходных данных

Для каждого теста в отдельной строке выведите «Yes» (без кавычек), если возможно всех путешествующих гусениц отправить в гости, и «No» (без кавычек) иначе.

Пример

стандартный ввод	стандартный вывод
4	Yes
2	No
10 20	No
1000000000 1000000000	Yes
4 4	
1 2	
2	
10 20	
1000000000 1000000000	
6 6	
1 2	
4	
0 9 13 15	
2 1 3 2	
6 6 1 1	
1 2 2 1	
4	
0 9 13 15	
2 1 3 2	
6 4 1 1	
1 2 2 1	