

Задача А. Планирование заданий

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

Имеется некоторое множество заданий и один исполнитель. На выполнение одного задания уходит единица времени. Задания можно выполнять начиная с момента времени 0. У каждого задания есть две характеристики: d_i и w_i . Если задание не было выполнено к моменту времени d_i , взимается штраф в размере w_i . Требуется минимизировать суммарный штраф.

Формат входных данных

Первая строка входного файла содержит натуральное число n — количество заданий ($1 \leq n \leq 100000$). Следующие n строк содержат по два натуральных числа, разделенных пробелом — d_i и w_i ($0 \leq d_i, w_i \leq 10^9$).

Формат выходных данных

Выведите одно число — минимальный суммарный штраф.

Пример

стандартный ввод	стандартный вывод
2 1 1 1 2	1

Задача В. Паросочетание максимального веса

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

Дан двудольный граф. Количество вершин в левой и правой доле совпадает и равно n . У каждой вершины левой доли есть вес, i -й вершине соответствует вес w_i . Вес паросочетания, ребрам которого инцидентны вершины левой доли a_1, a_2, \dots, a_k есть $\sqrt{\sum_{i=1}^k w_{a_i}^2}$. Требуется найти паросочетание максимального веса.

Формат входных данных

Первая строка входного файла содержит натуральное число n — количество вершин в обеих долях ($1 \leq n \leq 1000$). Вторая строка входного файла содержит n целых чисел w_1, w_2, \dots, w_n ($1 \leq w_i \leq 1000$). Следующие n строк содержат описания ребер, инцидентных соответствующей вершине левой доли. Формат описания: количество ребер, затем номера вершин правой доли, разделенные пробелом. Суммарное количество ребер не превосходит 200000.

Формат выходных данных

Выведите n чисел — для каждой вершины левой доли выведите номер вершины правой доли, с которой ее надо взять в паросочетание. Если вершина не входит в паросочетание, выведите 0.

Пример

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

Задача С. Проверка

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

Дано некоторое семейство множеств $S \subset 2^X$. Требуется проверить, может ли S быть семейством независимых множеств некоторого матроида.

Формат входных данных

Первая строка входного файла содержит два целых числа n и m — мощность множеств X и S соответственно ($1 \leq n \leq 10$, $0 \leq m \leq 2^n$). Каждая из следующих m строк содержит описание элемента множества S . Формат описания: количество элементов в подмножестве, затем через пробел номера этих элементов. Элементы множества X занумерованы, начиная с единицы.

Формат выходных данных

Выведите «YES», если S может быть семейством независимых множеств некоторого матроида, и «NO» иначе.

Примеры

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

Задача D. Ремонт дорог

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

Славная страна Матроидландия состоит из N городов, соединенных M шоссе. Каждое шоссе соединяет два различных города, и по каждому шоссе можно перемещаться в обе стороны. Некоторые пары городов могут быть соединены более чем одной дорогой. Правительство Матроидландии решило произвести модернизацию дорожной системы в стране. В стране есть несколько компаний, занимающихся ремонтом дорог. Они уже договорились, кто что будет ремонтировать, таким образом, для каждого шоссе указана компания, которая будет его ремонтировать.

Чтобы поддержать малый бизнес в развивающейся стране, Правительство решило, что каждая компания может ремонтировать не более, чем одно шоссе. Так как жители Матроидландии не очень любят думать, между любыми двумя городами должен существовать максимум один путь по отремонтированным дорогам. Ваша задача состоит в том, чтобы определить максимальное число шоссе, которые можно отремонтировать, чтобы все были довольны.

Формат входных данных

Первая строка входного файла содержит два числа — N и M ($1 \leq N \leq 100$, $0 \leq M \leq 5000$). Следующие M строк содержат описания шоссе. Каждое описание состоит из двух номеров городов u и v , которые соединяет шоссе, и номера c компании, которая это шоссе собирается ремонтировать ($1 \leq v, u \leq N$, $1 \leq c \leq 200$).

Формат выходных данных

На первой строке выведите единственное число K — искомое максимальное количество шоссе. На следующей строке выведите номера этих шоссе в любом порядке. Если существует несколько решений, выведите любое.

Примеры

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

Задача Ebonus. Рюкзак-матроид

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.5 секунд
Ограничение по памяти: 256 мегабайт

Одна из версий задачи о рюкзаке формулируется следующим образом: дано n предметов, i -й из них имеет вес w_i и стоимость v_i , требуется выбрать множество предметов с суммарным весом, не превышающим c (вместимости рюкзака) и максимальной возможной суммарной стоимостью. Известно, что задача о рюкзаке является NP-полной. Для нее существуют решения с помощью динамического программирования, но они обычно имеют время работы, линейное относительно суммы весов предметов и вместимости рюкзака, которое, таким образом, не является полиномиальным относительно размера входных данных.

Однако бывают случаи, когда задачу о рюкзаке можно так или иначе решить жадно. Один важный пример такого класса задач — если набор весов таков, что множество решений задачи образует матроид.

Матроидом называется пара $\langle X, \mathcal{I} \rangle$, где X — конечное множество, а \mathcal{I} семейство помножеств X , которые называют независимыми. При этом \mathcal{I} должно удовлетворять следующим трем свойствам:

1. $\mathcal{I} \neq \emptyset$;
2. Если $A \in \mathcal{I}$ и $B \subset A$, то $B \in \mathcal{I}$;
3. Если $A, B \in \mathcal{I}$ и $|A| > |B|$, то найдется такой $x \in A \setminus B$, что $B \cup \{x\} \in \mathcal{I}$.

Например, ребра неориентированного графа и семейство их ациклических подмножеств образуют матроид.

Рассмотрим предметы с весами w_1, w_2, \dots, w_n . Пусть X представляет собой множество целых чисел от 1 до n . Будем называть подмножество $\{i_1, i_2, \dots, i_k\}$ независимым, если $w_{i_1} + w_{i_2} + \dots + w_{i_k} \leq c$. Проверьте, образует ли получившаяся пара матроид.

Формат входных данных

Первая строка входного файла содержит число n ($1 \leq n \leq 50$). Вторая строка содержит n целых чисел w_1, w_2, \dots, w_n ($1 \leq w_i \leq 100$). Третья строка содержит число c ($\min w_i \leq c \leq \sum w_i$).

Формат выходных данных

Выведите «YES», если множество решений задачи о рюкзаке образует матроид. В противном случае выведите «NO».

Во втором случае выведите на второй строке число 2 или 3 — номер свойства, которое нарушается. Следующие две строки должны содержать контрпример к указанному свойству. Первая из строк должна описывать множество A , а вторая — B . Описание множества должно состоять из числа элементов в множестве и затем списка входящих в него предметов.

Примеры

стандартный ввод	стандартный вывод
3 1 2 3 4	YES
3 3 4 5 7	NO 3 2 1 2 1 3