

Задача А. Проверка

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

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 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.

Примеры

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

Задача С. Планирование заданий

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

Примеры

стандартный ввод	стандартный вывод
2 1 1 1 2	1

Задача D. Циклы

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

Циклом в матроиде называется множество $C \subset X$, которое не является независимым, а любое его собственное подмножество является независимым.

Дано некоторое семейство множеств $S \in 2^X$. Известно, что это множество циклов некоторого матроида. Кроме того, у каждого элемента множества X есть свой вес. Вес подмножества X есть сумма весов элементов, принадлежащих ему. Требуется найти базу максимального веса.

Формат входных данных

Первая строка входного файла содержит два натуральных числа n и m — мощность множеств X и S соответственно ($1 \leq n \leq 20$). Вторая строка входного файла содержит n чисел w_1, w_2, \dots, w_n ($1 \leq w_i \leq 1000$). Здесь элементы множества X занумерованы начиная с единицы и w_i — вес i -го элемента множества X . Каждая из следующих m строк содержит описание элемента множества S . Формат описания: количество элементов в подмножестве, затем через пробел номера этих элементов.

Формат выходных данных

Выведите одно число — вес максимальной базы.

Примеры

стандартный ввод	стандартный вывод
3 1 10 20 30 3 1 3 2	50

Задача Е. Peep Blue

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

Суперкомпьютер Peep Blue умеет играть в игры. Однако Серёжа не побоялся принять его вызов.

Правила, предложенные компьютером, таковы. Есть n кучек камней. Игроки ходят по очереди, начинает Серёжа. На первом ходу каждому из игроков разрешается убрать любой набор кучек (в том числе пустой, но не все). На втором и следующих ходах каждый из игроков может взять любое ненулевое число камней из любой кучки. Проигрывает тот, кто не может сделать ход.

Ваша задача — определить, может ли Серёжа одержать победу, а также какой первый ход ему следует сделать. Среди всех первых ходов, приносящих победу, выберите такой, который оставит в игре как можно больше камней: это должно усложнить задачу Peep Blue.

Формат входных данных

В первой строке ввода записано целое число n — количество кучек камней ($1 \leq n \leq 100\,000$). В следующей строке записаны n натуральных чисел, a_i , не превышающих $2^{30} - 1$, — количества камней в кучках.

Формат выходных данных

Если Серёжа не может одержать победу, выведите единственное число -1 . Иначе выведите максимально возможное суммарное число камней, которое можно оставить.

Примеры

реер.in	реер.out
4 1 2 3 4	9

Задача F. Рюкзак-матроид

Имя входного файла: knapsack.in
Имя выходного файла: knapsack.out
Ограничение по времени: 2 секунды
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 подмножеств 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

knapsack.in	knapsack.out
3 1 2 3 4	YES
3 3 4 5 7	NO 3 2 1 2 1 3