

Задача А. Расстановка книг

Имя входного файла: `input.txt`
 Имя выходного файла: `output.txt`
 Ограничение по времени: 3 секунды
 Ограничение по памяти: 256 мегабайт

Шёл 2116 год. Вася наконец-то получил на почте долгожданную юбилейную книгу «Барри Тоттер-100». Оказалось, что предыдущие книги про Барри стоят на полке шкафа не по порядку. Более того, книги других коллекционных серий также стоят не по порядку, но каждая на своей полке. Вася решил, что настало время расставить книги как положено (с ним это случается каждый раз, когда он получает новую книгу). Однако тратить много сил Вася не хочет, поэтому он решил что будет менять местами книги только стоящие рядом на одной полке, причем не более одного раза для каждого места. Помогите Васе написать программу, которая проверит, возможно ли поменять книги так, чтобы они были упорядочены по возрастанию номеров на каждой полке.

Формат входных данных

Первая строка входного файла содержит количество полок в шкафу $1 \leq n \leq 10$. Далее описываются книги на каждой полке. В i -ой строке первое число t_i равно количеству книг на i -ой полке, далее идут t_i номеров книг на i -ой полки в том порядке, в котором они стоят слева направо перед уборкой ($2 \leq t_i \leq 10^5$). Известно, что номера книг на полке – различные числа от 1 до t_i .

Формат выходных данных

В выходной файл нужно вывести n строк – по одной для каждой полки. Если переставить книги для данной полки возможно, то выведите строку «YES» (без кавычек), в противном случае выведите строку «NO» (без кавычек).

Примеры

<code>input.txt</code>	<code>output.txt</code>
1 3 2 3 1	YES
2 4 1 4 2 3 3 3 2 1	YES NO

Замечание

В первом тесте книги можно расставить по порядку, если поменять сначала книги с номерами 1 и 3, затем с номерами 2 и 1. Во втором примере книги на первой полке можно упорядочить, передвигая книгу с номером 4 два раза вправо. На второй полке можно менять местами книги только, стоящие на местах 1 и 2 или на местах 2 и 3, причём каждая замена разрешена только один раз. Нетрудно убедиться, что, используя эти две перестановки, расставить книги по порядку не получится.

Задача В. Обслуживание дорог

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

В Берляндии с нового года решили бороться с коррупцией. Начать борьбу решено с дорожной сферы. Между некоторыми городами страны есть дороги. Каждая дорога соединяет ровно два города Берляндии и имеет два направления. Известно, что сеть дорог в целях экономии построена так, что нет дублирующих маршрутов, т.е. нельзя выехать из одного города и проехав по дорогам, используя каждую не более одного раза, вернуться в тот же самый город. При этом из любого города можно добраться по дорогам до любого другого. Президент Берляндии потребовал, чтобы за каждой дорогой была закреплена ровно одна компания, которая будет обслуживать эту дорогу – чинить, убирать, наносить разметку и т.д. Чтобы всё было честно, было решено распределить дороги между компаниями так, чтобы ни у одной компании не было двух дорог, выходящих из одного и того же города. Помогите определить, какой наименьшее число компаний нужно привлечь, чтобы между ними можно было распределить дороги Берляндии по новому требованию президента.

Формат входных данных

В первой строке входного файла содержится два целых числа n и m – число городов и число дорог в Берляндии, соответственно ($2 \leq n \leq 10^5$, $1 \leq m \leq 10^5$). В следующих m строках описываются дороги: в i -ой строке содержится два различных числа a и b – номера городов, которые соединяет i -ая дорога ($1 \leq a, b \leq n$).

Формат выходных данных

В первой строке выходного файла нужно вывести число k – наименьшее число компаний, которые будут обслуживать дороги Берляндии. В следующей строке нужно вывести m чисел через пробел, i -ое число обозначает номер компании, которая будет обслуживать i -ую дорогу, номер компании должен быть числом от 1 до k . Если вариантов распределить дороги между k компаниями несколько – выведете любой.

Пример

input.txt	output.txt
3 2 1 2 2 3	2 2 1

Замечание

В примере три города соединены двумя дорогами. Так как обе дороги выходят из одного города, то одной компании будет недостаточно, с другой стороны, если дороги отдать на обслуживание разным компаниям, то требование будет выполнено.

Задача С. Футбольный беспорядок

Имя входного файла: `input.txt`
 Имя выходного файла: `output.txt`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 256 мегабайт

На футбольной арене Байтландии уже не первую неделю происходит настоящий футбольный беспорядок — неисправно цифровое табло. Одни пиксели экрана повреждены и теперь постоянно горят, другие — «выгорели», и их включение невозможно.

Несмотря на такую плачевную ситуацию, администрации стадиона нужно выполнять свои обязательства: транслировать рекламные ролики спонсоров, отображать информацию во время матча и др. Очевидно, что при описанных неисправностях далеко не каждое изображение может быть выведено на экран без помех. Помогите руководству стадиона для некоторого набора изображений определить, какие из них можно вывести на экран в неизменном виде.

Формат входных данных

В первой строке входного файла через пробел заданы два целых числа N и M — число строк и число столбцов экрана, соответственно ($1 \leq N, M \leq 100$). В следующих N строках находится описание пикселей экрана. Каждая строка содержит M символов. Символ «+» означает, что данный пиксель находится в рабочем состоянии, символ «*» показывает, что данный пиксель постоянно горит, символ «.» — постоянно не горит.

В следующей строке записано целое число K — количество картинок, для которых нужно выполнить проверку ($1 \leq K \leq 20$). Далее следуют описания K картинок. Каждое описание состоит из N строк по M символов в каждой. Символ «*» означает, что данный пиксель в картинке должен гореть, символ «.» — не гореть.

Формат выходных данных

В выходной файл выведите K строк (по одной на каждую картинку) — результаты проверки каждой из картинок. В случае, если на табло можно вывести картинку, выведите «YES» (без кавычек). В противном случае, выведите «NO» (без кавычек).

Пример

input.txt	output.txt
<pre>6 10 ++.....++ .+++++++. .+*****. .+*****. .+++++++. ++.....++ 2 **.....** ..*.*.*.. ..*****. ..*****. ..*.*.*.. **.....** ******....*....*....*....*....</pre>	<pre>YES NO</pre>

Задача D. Петя и устный счёт

Имя входного файла: `input.txt`
 Имя выходного файла: `output.txt`
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 256 мегабайт

Школьник Петя, вдохновившись известными телешоу о необычных талантах, решил стать участником одного из них. Принимая во внимание свои успехи в математике, Петя решил активно развивать своё умение производить арифметические операции в уме. Достигнув невероятных результатов в этом деле, перед ним встала задача подготовки своего выступления на телевидении...

Посоветовавшись со своим учителем математики, Петя решил выступать со следующим номером. Ассистент по очереди называет вслух N целых положительных чисел. После оглашения всех N чисел, участник по порядку перечисляет те из них, которые образуют **длиннейшую** подпоследовательность исходной последовательности, каждый следующий элемент которой делится на предыдущий. Более формально, если изначально была названа последовательность a_1, \dots, a_N , то Петя должен назвать подпоследовательность a_{i_1}, \dots, a_{i_k} максимальной длины такую, что $a_{i_m} \mid a_{i_{m+1}}$ ($i_1 < \dots < i_k$).

Для тренировки номера требуется программа, которая будет следить за тем, что участник правильно находит требуемую длиннейшую подпоследовательность. Помогите Пете в написании такой программы.

Формат входных данных

В первой строке входного файла задано одно целое число N — количество чисел в последовательности ($1 \leq N \leq 10^5$). В следующей строке через пробел задано N целых чисел a_i — элементы исходной последовательности ($1 \leq a_i \leq 10^4$).

Формат выходных данных

В первой строке выходного файла выведите целое число K — максимальную длину подпоследовательности, удовлетворяющей заданному условию. В следующей строке выведите K целых чисел — индексы элементов подпоследовательности в порядке их расположения в последовательности. Если оптимальных ответов несколько, выведите любой.

Пример

input.txt	output.txt
6	5
1 2 2 6 4 16	1 2 3 5 6

Замечание

Подпоследовательностью называется набор элементов последовательности, в котором каждый следующий элемент находится **правее** предыдущего в исходной последовательности.

Задача Е. Амбар

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

В стране Полигонии по периметру границы расставлены сторожевые башни. Периодически для обмена информацией между башнями от одной к другой отправляется всадник с посланием. Поскольку лошадям приходится преодолевать большие расстояния, то было решено построить амбар, в котором будет храниться сено для лошадей, и откуда оно будет развозиться по разным местам для подкормки животных. Будущий амбар решили построить в том месте, где пересекается наибольшее число путей всадников, скачущих между башнями. Помогите найти жителям Полигонии точное место, где будет построен амбар.

Формат входных данных

В первой строке входного файла содержится единственное число n – число сторожевых башен в Полигонии ($4 \leq n \leq 50$). Далее в n строках описываются местоположения башен. В i -ой строке содержится два действительных числа x_i и y_i – координаты i -ой башни (предполагается, что башни представляются точками на карте) ($0 \leq x_i, y_i \leq 10^4$). Известно также, что Полигония представляется на карте выпуклым многоугольником, вершины которого – сторожевые башни. Башни перечисляются при обходе границы против часовой стрелки.

Формат выходных данных

В выходной файл в первой строке нужно вывести количество путей всадников, которые будут проходить через место с амбаром. Во второй строке необходимо вывести координаты будущего места под амбар через пробел – в этой точке должно пересекаться наибольшее число путей всадников. Предполагается, что всадники скачут от одной башни к другой всегда вдоль прямой линии. Если таких мест можно выбрать несколько, то выведите любое. Ответ будет засчитан, если он отличается от правильного менее, чем на 10^{-6} .

Пример

input.txt	output.txt
4	2
1 0	0.5000000 0.5000000
1 1	
0 1	
0 0	

Задача F. Шутка Ефима

Имя входного файла: `input.txt`
 Имя выходного файла: `output.txt`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 256 мегабайт

Ефим – большой любитель пошутить, и буквально на днях придумал новую великолепную шутку. Но вот незадача – оценить качество и стиль своего юмора у него никак не получается.

Понятно, что чем больше в шутке смешных слов, тем она считается более смешной. Поэтому Ефим раздобыл несколько смеховых словарей и теперь хочет понять, из какого из них он употребил наибольшее количество различных слов в своей шутке, это помогло бы ему определить стиль юмора и, возможно, выбрать этот словарь для дальнейших упражнений. Если таких словарей окажется несколько, то Ефима интересуют все подходящие варианты.

Но подойти к вопросу серьёзно у Ефима никак не получается, поэтому он попросил вас немного помочь ему в этом (т.е. сделать всю работу за него).

Формат входных данных

В первой строке входного файла содержится единственное целое число N – количество смеховых словарей ($1 \leq N \leq 100$). В следующих N строках описываются сами словари. В i -й строке содержится число a_i – количество слов в i -м словаре, далее перечисляются a_i слов, разделённых пробелом и состоящих из строчных букв латинского алфавита ($1 \leq a_i \leq 10^3$). Гарантируется, что все слова в рамках одного словаря различны.

Следующая строка содержит единственное число K – количество слов в шутке Ефима ($1 \leq K \leq 10^3$). Последняя строка содержит K слов, из которых состоит шутка Ефима. Слова разделены символом пробела и состоят из строчных букв латинского алфавита.

Гарантируется, что суммарная длина всех слов не превосходит 10^6 .

Формат выходных данных

В первой строке выходного файла требуется вывести два целых числа – количество максимально подходящих словарей (т.е. словарей, с которыми у шутки наибольшее количество общих различных слов) и количество совпавших слов соответственно. Во второй строке требуется вывести через пробел номера максимально подходящих словарей, упорядоченные по возрастанию.

Примеры

input.txt	output.txt
2 1 az 2 az za 3 az az a	2 1 1 2
2 1 hoh 2 hoh ho 3 hoh hoh ho	1 2 2

Задача G. Показатель и указатель

Имя входного файла: `input.txt`
Имя выходного файла: `output.txt`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Арнольд — умелый хакер, а помимо этого ещё и выдающийся математик, именно поэтому он постоянно путает понятия показатель и указатель. Но сегодня ему повезло. Показатель и указатель в его задаче оказались почти одним и тем же. Нужно лишь прибавить к известному Арнольду адресу смещение, равное значению показателя, как он тут же получит адрес, в котором лежит ваш пароль.

Чтобы найти этот показатель Арнольду потребуется решить математическую задачу: дано большое число $n = m_1 * m_2 * m_3 * \dots * m_M$, записанное в виде произведения небольших чисел m_i . Нужно представить число n в виде $n = a^p$, где a, p — некоторые целые числа, причем p — максимально для всех таких представлений. Как вы уже догадались, число p является искомым показателем.

Теперь вы обладаете той же информацией, что и Арнольд. Успеете ли вы опередить его и стереть свой пароль из этого места?

Формат входных данных

В первой строке входного файла содержится число M — количество множителей в произведении ($1 \leq M \leq 1000$). Во второй строке содержатся M целых чисел m_i — множители произведения ($1 \leq i \leq M, 1 \leq m_i \leq 10^6$).

Формат выходных данных

В выходной файл нужно вывести единственное целое число p — максимальный искомый показатель.

Примеры

<code>input.txt</code>	<code>output.txt</code>
3 9 4 4	2
3 3 9 81	7

Задача Н. Кадровые перестановки

Имя входного файла: `input.txt`
 Имя выходного файла: `output.txt`
 Ограничение по времени: 1 секунда
 Ограничение по памяти: 256 мегабайт

В канун Нового года в компании *BetGains* царит непередаваемая атмосфера: подготовка к празднованию идёт полным ходом, изредка прерываясь на рабочие моменты. Вернувшись из деловой поездки, генеральный директор компании понял: «Нужно срочно что-то менять», и первым делом взялся за кадровые перестановки.

Каждый сотрудник компании может иметь в своём непосредственном подчинении несколько (возможно, 0) сотрудников. При этом сотрудник *A* считается *начальником* сотрудника *B* **тогда и только тогда**, когда существует последовательность сотрудников, начинающаяся с сотрудника *A* и заканчивающаяся сотрудником *B*, где каждый сотрудник, кроме последнего, является непосредственным начальником следующего.

В силу несовершенства системы учёта кадров, в *BetGains* **возможна** ситуация, в которой два сотрудника являются начальниками друг-друга.

Директор *BetGains* хочет провести необычную кадровую перестановку, а именно: выбрать двух сотрудников, один из которых является непосредственным подчинённым другого, и поменять их ролями. Другими словами, если сотрудник *A* был непосредственным подчинённым сотрудника *B*, то после кадровых перемен сотрудник *B* станет непосредственным подчинённым сотрудника *A*.

На данный момент совет директоров компании интересуется следующий вопрос: какое максимальное количество сотрудников может иметь общего начальника после такой кадровой перестановки?

Формат входных данных

В первой строке входного файла задано единственное целое число N — количество сотрудников в компании ($1 \leq N \leq 400$). В следующих N строках заданы списки непосредственных подчинённых каждого из сотрудников. Каждое описание начинается целым числом k_i , равным количеству человек в непосредственном подчинении i -го сотрудника. Далее в этой же строке через пробел записаны k_i целых чисел a_{ij} — номера сотрудников в непосредственном подчинении сотрудника с номером i . Гарантируется, что $\sum k_i \leq 400$ и хотя бы для одного i число k_i ненулевое.

Формат выходных данных

В выходной файл нужно вывести 4 числа, разделённых пробелом. Первое число должно равняться максимально возможному количеству сотрудников, имеющих общего начальника после кадровой перестановки. Следующие два числа — это номера сотрудников, подлежащих кадровой перестановке (до перестановки первый из них должен быть непосредственным начальником второго). Последнее число должно быть равно номеру сотрудника, который будет общим начальником для максимально возможного количества сотрудников, после указанной кадровой перестановки. Если вариантов ответа существует несколько — выведите любой из них.

Примеры

input.txt	output.txt
3 1 2 1 3 0	2 1 2 2
4 1 2 0 2 2 4 0	3 1 2 3