Задача A. Кристаллы

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

В результате последних исследований в лаборатории «Аномальных материалов» был найден способ закрыть мост между мирами. На Земле были обнаружены новые образцы кристаллов. Внимательно изучив их, ученые пришли к выводу, что для каждого из них должно существовать отображение в любом параллельном измерении. Для проверки этой гипотезы на планету Зен был отправлен отряд, который доставил отображения земных кристаллов в лабораторию.

Дальнейшие исследования показали, что каждый из кристаллов, найденый на обеих планетах, характеризуется некоторым натуральным числом — мощностью излучения. Для закрытия моста нужно подобрать такой набор земных кристаллов, что его суммарная мощность окажется равной суммарной мощности их отображений, обнаруженных на планете Зен. Оборудование лаборатории позволяет клонировать найденные кристаллы, поэтому любой из них можно использовать несколько раз.

К сожалению, после неудачного опыта с кристаллом, открывшим проход, суперкомпьютеры отказались сотрудничать с учеными из лаборатории, но они согласны работать с вами! Помогите исследователям «Черной Мезы» спасти мир.

Формат входного файла

Первая строка входного файла содержит одно целое число n ($1 \le n \le 10\,000$) — количество типов кристаллов. Следущая строка содержит n чисел a_i ($0 \le a_i \le 1\,000$) — мощности кристаллов, найденных на планете Земля. Последняя строка содержит n чисел b_i ($0 \le b_i \le 1\,000$) — мощности отображений земных кристаллов, доставленных с планеты Зен.

Формат выходного файла

Если закрыть мост возможно, в первой строке выходного файла выведите число k ($1 \le k \le 10\,000$) — количество кристаллов, необходимое для закрытия моста. Во второй строке выведите k чисел — номера типов кристаллов, которые будут использоваться в эксперименте. Один и тот же тип кристалла можно использовать несколько раз.

Если закрыть мост невозможно, выведите -1.

Примеры

crystals.in	crystals.out
3 3 4 5 1 6 2	2 1 2
3 3 6 5 1 4 2	-1
2 1 4 3 2	4 1 1 2 2

Комментарий

Решения, работающие в случаях, в которых n не превышает 100, будут оцениваться в 60 баллов.

Задача B. Пёсик

Имя входного файла:	`doggy.in`
Имя выходного файла:	`doggy.out`
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Пёсик, принадлежащий Аликс, как и все песики, любит играть с мячиком. Но в этот раз вместо мячика ему попалась конструкция из металлических звеньев. Каждое звено в ней может быть соединено с некоторыми другими звеньями. Пёсику срочно необходимо с ней поиграть! Для этого Пёсику необходимо удалить несколько звеньев (возможно, ноль) и разорвать несколько (также, возможно, ноль) связей между звеньями, чтобы в итоге получилась замкнутая цепь из звеньев, в которой каждое звено соединено только с двумя соседними. Каждое звено в такой цепи должно быть использовано ровно один раз.

Пёсик хочет, чтобы получившаяся цепь была как можно длиннее. К счастью, изначальная конструкция такова, что вариантов сделать замкнутую цепь у него немного, а именно, не более трех. Помогите Пёсику!

Формат входного файла

В первой строке входного файла находятся два целых числа n и m ($1 \le n \le 200\,000$, $1 \le m \le 500\,000$) — количество звеньев и соединений между ними соответственно. В следующих m строках находятся описания соединений: в каждой строке находится по два числа u и v ($1 \le u, v \le n$), означающие, что звенья с номерами u и v соединены между собой. Никакое звено не соединено с собой, и никакие два звена не соединены более одного раза. Гарантируется, что из данной конструкции можно получить хотя бы одну замкнутую цепь. Также гарантируется, что имеется не более трех вариантов удаления соединений и звеньев, после применения которых остается замкнутая цепь из звеньев. Исходная конструкция является связной.

Формат выходного файла

В первой строке выходного файла выведите число k: длину максимальной замкнутой цепи, которую может получить Пёсик. В следующей строке выведите k чисел $a_1, a_2, \ldots, a_k$ — номера звеньев в этой цепи. Все a_i должны быть различны, и звенья с номерами a_1 и a_2, a_2 и a_3, …, a_{k-1} и a_k, а также a_k и a_1 должны быть соединены в изначальной конструкции.

doggy.in	doggy.out
4 5 3 2 2 4 4 3 2 1 1 3	4 3 1 2 4

Комментарий

Решения, работающие в случаях, когда есть единственный вариант получить замкнутую цепь, будут оцениваться в 20 баллов.

Решения, работающие в случаях, когда есть не более двух вариантов получить замкнутую цепь, будут оцениваться в 40 баллов.

Решения, работающие в случаях, когда есть не более трех вариантов получить замкнутую цепь, но никакие две цепи, которые можно получить, не имеют общих звеньев, будут оцениваться в 60 баллов.

Задача C. Накорми хедкраба

Имя входного файла:	`gonarch.in`
Имя выходного файла:	`gonarch.out`
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайта

Широкоизвестный житель Зена Гонарч на самом деле не только ужасный монстр, но и любящий отец. Как и любой отец, Гонарч считает своим долгом накормить всех своих детей. Но, так как их слишком много, большой папочка попросил взрослых хедкрабов помочь ему. Он думает, что помощники разделили малышей примерно поровну, и выдает всем кормящим хедкрабам одинаковое число пищи. К сожалению, хедкрабы достаточно глупы, и берут себе столько малышей, сколько взбредет им в голо..., стоп, ведь у хедкрабов нет головы. Личинки рождаются непрерывно, и если их не возьмется кормить какой-то хедкраб, то они умрут. Эта ситуация мало кого волнует, ведь их все равно очень много.

Получая n единиц пищи, хедкраб, уполномоченый кормить группу из k личинок, выдает каждой личинке максимальное возможное число единиц пищи такое, что все личинки получают поровну. Всю остальную еду съедает сам хедкраб.

Гонарч прознал, что хедкрабы едят еду, предназначенную личинкам, и, разумеется, не хочет кормить взрослых хедкрабов, которые могут добывать еду самостоятельно. Поэтому, прежде чем раздавать еду, Гонарч хочет узнать сколько еды достанется взрослым хедкрабам. Вам нужно написать программу, которая бы могла отвечать на запросы Гонарч. Программа должна обрабатывать следующие события:

- $+\ k$ — появился хедкраб, который взялся кормить группу из только-что родившихся k личинок.
- - k — один из хедкрабов, взявшихся кормить группу из k личинок, бросил их и ушел охотиться за мозгами людей. После этого личинки сразу умирают и эта группа пропадает.
- ? n — Гонарч собирается выдать каждому кормящему хедкрабу n единиц еды. Программа должна вывести сколько единиц еды съедят сами кормящие хедкрабы.

К тому времени как Гонарч обратился к вам, возможно некоторые хедкрабы уже взяли себе группы по несколько малышей. Гонарч сообщит Вам эту информация в начале работы программы.

Формат входного файла

В первой строке задано число m — количество различных размеров групп, существовавших до начала работы программы ($0 \le m \le 10^5$). В следующих m строках заданы описания существовавших групп. В $i+1$ —ой строке заданы два числа g_i и c_i, размер группы и количество таких групп соответственно. Гарантируются, что все g_i различны ($1 \le g_i, c_i \le 10^5$).

В $m+2$ строке входного файла задано количество запросов t ($1 \le t \le 10^5$). В следующих t строках заданы запросы по одному в строке. Запросы могут быть одного из следующих видов:

- Запрос появления нового кормящего хедкраба, вида $+\ k$ ($1 \le k \le 10^5$), где k — количество личинок, за которые отвечает этот хедкраб.
- Запрос исчезновения кормящего хедкраба, вида - k, где k ($1 \le k \le 10^5$) — количество подопечных ушедшему хедкрабу личинок. Гарантируется, что на момент выполнения этого запроса существует хотя бы один хедкраб с таким числом подопечных личинок.
- Запрос ? n, где n ($1 \le n \le 10^5$) — количество единиц еды, которое получит каждый кормящий хедкраб.

Формат выходного файла

На каждый запрос о выдаче кормящим хедкрабам еды нужно в отдельной строке вывести ответ на этот запрос. Ответ должен состоять из одного числа, равного количеству единиц еды, которые достанутся кормящим хедкрабам.

Примеры

gonarch.in	gonarch.out
1 2 1 7 + 3 + 2 ? 5 - 2 ? 7 - 2 ? 4	4 2 1

Комментарий

Решения, работающие в случаях, в которых $t \le 10^3$ будут оцениваться в 20 баллов.

Решения, работающие в случаях, в которых $n = 0$ и $t \le 10^4$ будут оцениваться в 20 баллов.

Решения, работающие в случаях, в которых нет запросов на удаление, и не найдется двух хедкарбов с одинаковыми размерами групп, будут оцениваться в 20 баллов.

Задача D. Чертеж

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

Убегая от очередного недружелюбного существа из параллельного измерения, Гордон Фримен укрылся в лаборатории, где раньше изучали поведение различных кристалов, доставленных из измерения Зен. Когда он понял, что ему не угражает никакая опасность, Фримен осмотрелся и увидел интересное оборудование, которое предназначалось для сверхточного рисования сложных схем. Однако, оно не было расчитано на рисования обычных чертежей, и не могло нарисовать некоторые из них в силу своего устройства. На компьютере, который был в лаборатории, Фримен нашел чертежи всех помещений федерального исследовательского центра, наличие которых в распечатенном виде очень сильно увеличило бы его шансы на выживание. Нетрудно догадаться, что он решил сделать!

Само устройство по рисованию схем выглядело следующим образом. В некоторой точке A_0 плоскости чертежа была закреплена «рука», состоящая из n последовательно соединенных с помощью шарниров металических сегментов. Обозначим длину первого сегмента (того, который прикреплен к точке A_0) как l_1, следующего l_2, ..., последнего l_n. Обозначим точки соединения сегментов с номерами i и $i+1$ как A_i, а конец последнего сегмента A_n. В точке A_n закреплен инструмент, который позволяет рисовать точку на чертеже. Устройство управления может разместить шарнирные соединения в точках A_i в плоскости чертежа, если для всех $i < n$ справедливо, что $|A_iA_{i+1}| = l_i$.

Чтобы нарисовать чертежи, Фримену необходимо узнать, все ли необходимые точки сможет нарисовать это устройство. Также для каждой точки ему необходимо узнать координаты всех A_i. Помогите ему в этом.

Формат входного файла

В первой строке входного файла находятся четыре целых числа x_0, y_0, x_n, y_n ($-10\,000 \le x_0, y_0, x_n, y_n \le 10\,000$) — координаты точки A_0, к которой прикреплен первый сегмент чертежного устройства, а также точки A_n, которую необходимо нарисовать.

В следующей строке дано целое число n ($1 \le n \le 10^3$) — количество сегментов в устройстве рисования. В следующей строке находятся n целых чисел l_i ($1 \le l_i \le 1\,000$) — длины сегментов.

Формат выходного файла

В первой строке выходного файла выведите `YES`, если можно нарисовать точку A_n и `NO` в противном случае. Если точку нарисовать можно, то в следующих $n+1$ строках выведите координаты A_i для всех i от нуля до n. Ответ будет засчитан, если расстояние между выведенными вами точками A_i и A_{i+1} отличается от l_i не более, чем на 10^{-3}.

Примеры

painting.in	painting.out

`0 0 6 0`
`2`
`5 5`
	`YES`
	`0.000 0.000`
	`3.000 4.000`
	`6.000 0.000`

Комментарий

Количество тестов, в которых $\sum_1^n l_i$ не превышает расстояния от точки A_0 до точки A_n, составляет 20 процентов от общего числа тестов.

Количество тестов, в которых $n \le 2$, составляет 40 процентов от общего числа тестов.