

Задача 3. ПЕСНЬ

имя входного файла: input.txt
 имя выходного файла: output.txt
 ограничение времени на каждом тесте: 10 сек
 баллы: 34 балла

*...Звуки умертвие,
 Музыку я разъял, как труп.
 Проверил
 Я алгеброй гармонию...*
 А. С. Пушкин

В традиционной музыке используются музыкальные звуки из некоторого набора, именуемого *звукорядом*. Звуки звукоряда принято группировать в *октавы*, в каждой из которых 12 звуков. Порядковый номер звука в пределах одной октавы назовем *нотой*. Таким образом, каждый звук можно задать парой чисел – номером октавы и номером ноты. Номер октавы K – произвольное целое число, номер ноты N принимает значение из интервала $[0,12]$. Звуки можно обозначать этими двумя числами, записанными рядом без пробелов (второе число всегда двузначное). Эту запись назовем *кодом* Q . Например, $Q = -108$ для (-1) -й октавы, восьмой ноты. Значение Z , определяемое по формуле $Z = K \cdot 12 + N$ назовем *абсолютным номером* Z звука в звукоряде (для приведенного выше примера $Z = -4$).

Набор всех звуков, ноты которых принадлежат заданному подмножеству P номеров нот, назовем *гармонией* G . Это означает, что любой звук с абсолютным номером $Z = K \cdot 12 + n$, где n – номер ноты из P , принадлежит этой гармонии при любом значении K . Отсюда следует, что гармония однозначно определяется указанием P . Две гармонии назовем *эквивалентными*, если при прибавлении некоторого одного и того же целого числа ко всем абсолютным номерам звуков первой гармонии получаются все элементы второй гармонии. Ограничимся рассмотрением только таких наборов гармоний, в которые наряду с каждой из гармоний входят и все эквивалентные ей. Для описания набора такого вида достаточно указать из каждой совокупности эквивалентных по одной гармонии G_i или соответствующему ей подмножеству P_i . *Базой* B набора гармоний назовем совокупность всех таких P_i .

Всякую совокупность одновременно звучащих звуков (не менее двух) будем называть *аккордом* A . Для некоторого заданного набора гармоний назовем *гармонией аккорда* A такую гармонию G из него, что все звуки аккорда A принадлежат G .

Будем говорить, что некоторый звук *в тему* для некоторого аккорда, если этот звук принадлежит хотя бы одной гармонии из множества всех гармоний этого аккорда.

Последовательное звучание произвольных звуков назовем *мелодией*. Каждому звуку мелодии может быть сопоставлен аккорд в порядке исполнения мелодии. Будем считать мелодию *благозвучной* для этой последовательности аккордов, если каждый ее звук оказывается в тему для соответствующего ему аккорда. *Кучерявостью* мелодии назовем сумму модулей разностей абсолютных номеров Z последовательно исполняемых звуков данной мелодии.

Задание

Пусть даны: база B набора гармоний, последовательность аккордов A и начальный звук Q . Требуется написать программу, находящую наименее кучерявую из всех благозвучных мелодий, начинающихся с этого звука.

Формат входных данных

В первой строке записано целое число M — количество заданных элементов в базе набора гармоний ($1 \leq M \leq 200$). Далее следуют M строк, каждая из которых содержит описание одного из элементов в базе набора гармоний в виде последовательности составляющих ее номеров нот, записанных через пробел. В следующей строке находится целое число L — количество заданных аккордов ($1 \leq L \leq 200$). Каждая из последующих L строк содержит описание одного из аккордов в виде последовательности кодов составляющих его звуков, записанных через пробел. Описания аккордов следуют в порядке их исполнения. Последняя строка входного файла содержит код начального звука Q . Все значения кодов звуков записываются тремя цифрами.

Формат выходных данных

В первую строку выходного файла следует вывести минимально возможную кучерявость среди всех мелодий, удовлетворяющих описанным выше требованиям. Оставшиеся строки должны содержать L целых чисел — коды Q звуков, составляющих соответствующую мелодию. Если вариантов мелодий несколько, нужно вывести любую из них. Для заданных во входном файле данных всегда будет существовать хотя бы одна благозвучная мелодия.

Пример входного файла

```
3
1 5 8 11
1 5 8 12
1 6 8
5
101 106
010 112
-101 004
201 202
110 111
102
```

Пример выходного файла

```
4
102 104 104 106 106
```