A. Плейлисты

|  |  |
| --- | --- |
| Ограничение времени | 1.5 секунд |
| Ограничение памяти | 256Mb |
| Ввод | стандартный ввод или input.txt |
| Вывод | стандартный вывод или output.txt |

Костя успешно прошел собеседование и попал на стажировку в отдел разработки сервиса «Музыка».

Конкретно ему поручили такое задание — научиться подбирать плейлист для группы друзей, родственников или коллег. При этом нужно подобрать такой плейлист, в который входят исключительно нравящиеся всем членам группы песни.

Костя очень хотел выполнить это задание быстро и качественно, но у него не получается. Помогите ему написать программу, которая составляет плейлист для группы людей.

Формат ввода

В первой строке расположено одно натуральное число n(1≤n≤2⋅105), где n – количество человек в группе.

В следующих 2⋅n строках идет описание любимых плейлистов членов группы. По 2 строки на каждого участника.

В первой из этих 2-х строк расположено число ki — количество любимых треков i-го члена группы. В следующей строке расположено ki строк через пробел — названия любимых треков i-го участника группы.

Каждый трек в плейлисте задан в виде строки, все строки уникальны, сумма длин строк не превосходит 2⋅106. Строки содержат большие и маленькие латинские буквы и цифры.

Формат вывода

Выведите количество, а затем сам список песен через пробел — список треков, которые нравятся каждому участнику группы. Ответ необходимо **отсортировать**в лексикографическом порядке!

Пример 1

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 1  2  GoGetIt Life | 2  GoGetIt Life |

Пример 2

| **Ввод**  Скопировать ввод | **Вывод**  Скопировать вывод |
| --- | --- |
| 2  2  Love Life  2  Life GoodDay |  |

using System;

using System.Collections.Generic;

using System.Linq;

class Program

{

static void Main()

{

int N = Convert.ToInt32(Console.ReadLine());

HashSet<string> result = new HashSet<string>();

for (int i = 0; i < N; i++)

{

int k = Convert.ToInt32(Console.ReadLine());

string[] songs = Console.ReadLine().Split();

if (i == 0)

{

// Array.Sort(songs);

for (int j = 0; j < k; j++)

{

result.Add(songs[j]);

}

}

else

{

HashSet<string> current = new HashSet<string>();

for (int j = 0; j < k; j++)

{

current.Add(songs[j]);

}

foreach (string song in result)

{

if (!current.Contains(song))

{

result.Remove(song);

}

}

}

}

Console.WriteLine(result.Count);

List<string> sortedList = result.ToList();

sortedList.Sort(); // сортировка в порядке возрастания

// Выводим отсортированные элементы

foreach (string number in sortedList)

{

Console.Write(number + " ");

}

}

}