

Задача А. Новая дорожная система

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Король некоторой страны N решил полностью перестроить дорожную систему. В стране проживает n человек, пронумерованных от 1 до n . Можно проводить двустороннюю дорогу от домика жителя этой страны с номером a к домику другого жителя этой страны с номером b . Между каждой парой жителей не может быть построено больше одной дороги. Конечно система дорог должна быть связной, то есть от любого домика можно дойти до другого передвигаясь по дорогам. Для экономии средств было решено построить $n - 1$ дорог, то есть, система дорог должна образовывать дерево.

Но не все так просто, ведь не зря король обратился именно к вам за помощью. В стране есть m тайных обществ, известных только королю, каждое из которых является непустым подмножеством её жителей. Поскольку он не хочет вступать в конфликт ни с одним из этих обществ, он хочет построить дорожную систему, которая будет удовлетворять следующему условию: все домики жителей каждого тайного общества должны образовывать связное поддерево. Будем считать, что некоторое подмножество вершин дерева образует связное поддерево тогда и только тогда, когда выполнено условие: если оставить только вершины этого подмножества и ребра дерева, соединяющие только вершины из подмножества, то полученный граф будет связным.

Помогите королю определить, возможно ли построить такую дорожную систему или нет, и если возможно, постройте её.

Формат входных данных

Каждый тест содержит один или несколько тестовых случаев.

На первой строке входного файла находится единственное число t ($1 \leq t \leq 2000$) — количество тестовых случаев.

В следующих строках следуют описания этих тестовых случаев в следующем формате.

На первой строке находятся два целых числа n и m ($1 \leq n, m \leq 2000$) — количество жителей в стране и количество тайных обществ.

В следующих m строках находится описание тайных обществ. В i -й из этих строк находится строка s_i длины n , состоящая из символов «0» и «1». Житель с номером j принадлежит i -му тайному обществу тогда и только тогда, когда $s_{ij} = 1$. Гарантируется, что в строке s_i есть хотя бы один символ «1» для любого $1 \leq i \leq m$.

Гарантируется, что сумма всех n для всех тестовых случаев не превосходит 2000 и сумма всех m для всех тестовых случаев не превосходит 2000.

Формат выходных данных

Выведите ответы для всех тестовых случаев в том порядке, в котором они заданы во входных данных, в следующем формате.

Если дорожную систему построить невозможно, в единственной строке выведите «NO» (без кавычек).

В противном случае в первой строке выведите «YES» (без кавычек).

В следующих $n - 1$ строках выведите описание дорожной системы: в каждой строке должно находиться по два целых числа a, b , которые будут означать, что нужно провести дорогу между домиками жителей a и b ($1 \leq a, b \leq n, a \neq b$).

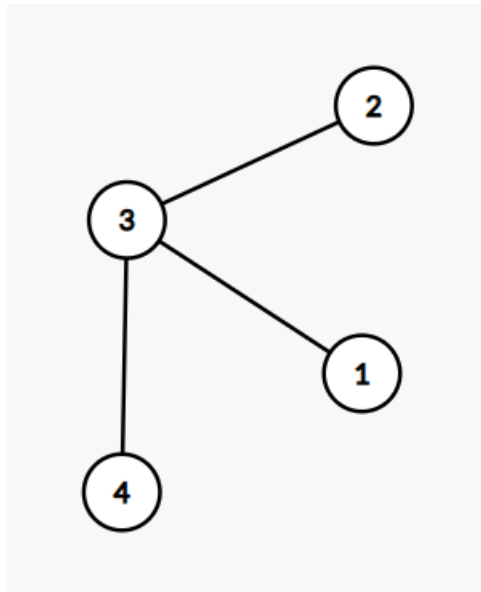
Проведенные дороги должны образовывать дерево, и домики каждого общества должны образовывать связное поддерево.

Пример

стандартный ввод	стандартный вывод
2	YES
4 3	1 3
0011	2 3
1110	3 4
0111	NO
3 3	
011	
101	
110	

Замечание

В первом тестовом случае можно построить такую дорожную систему:



Легко убедиться, что для каждого тайного общества все домики его жителей образуют связное поддерево. Например, во 2-м тайном обществе состоят жители с номерами 1, 2, 3. Они образуют связное поддерево, так как если оставить только домики 1, 2, 3 и дороги между ними, то останутся две дороги: между 1 и 3 и между 2 и 3, то есть связный граф.

Во втором тестовом случае можно убедиться, что не существует ни одной дорожной системы, подходящей королю.

Задача В. Электрическая схема

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Паша — начинающий техник, но уже поставил себе большую цель собрать собственный компьютер. Первая непростая задача — научиться собирать электрическую схему.

Схема, которую собрал Паша, состоит из несколько проводов. Каждый провод — это отрезок, который соединяет две точки на плоскости с целыми координатами, лежащими в диапазоне $[1, 10^9]$.

В схеме есть провода двух цветов:

- красные провода: эти провода должны иметь вид горизонтального отрезка, то есть если провод соединяет две точки (x_1, y_1) и (x_2, y_2) , то выполнено, что $y_1 = y_2$;
- синие провода: эти провода должны иметь вид вертикального отрезка, то есть если провод соединяет две точки (x_1, y_1) и (x_2, y_2) , то выполнено, что $x_1 = x_2$.

Обратите внимание, что если провод соединяет две одинаковые точки, то он может быть как красным, так и синим. Также в Пашиной схеме никакие два провода одного цвета не могут пересекаться, то есть любые два отрезка проводов одного цвета не могут содержать общих точек.

Недоработка Пашиной схемы состоит в том, что его провода не были изолированы, и поэтому в точках пересечения проводов разных цветов возникли искры, которые Паша увидел. Он записал все точки, в которых он увидел искру. У него получилось множество из n различных точек. После чего он разобрал схему и пошёл спать.

Утром, когда Паша увидел на листочке множество из n точек, в которых он увидел искру, ему стало интересно, сколько проводов он использовал, собрав эту схему. К сожалению, он ничего не запомнил, поэтому он решил узнать, какое минимальное количество проводов он мог использовать в своей схеме. Помогите ему узнать это число, а также расположить эти провода так, чтобы в получившейся схеме искры возникли в тех же самых точках.

Формат входных данных

В первой строке находится одно целое число n ($1 \leq n \leq 1000$) — количество точек, в которых Паша увидел искру.

В следующих n строках находится по два целых числа x и y ($1 \leq x, y \leq 10^9$) — координаты очередной точки. Гарантируется, что все точки различны.

Формат выходных данных

Выведите описание электрической схемы в следующем формате:

Сначала выведите h — количество горизонтальных красных проводов ($0 \leq h$). В следующих h строках выведите по 4 целых числа x_1, y_1, x_2, y_2 — координаты двух точек (x_1, y_1) и (x_2, y_2) , которые соединяет очередной красный провод. Поскольку отрезки горизонтальные, должно быть выполнено $y_1 = y_2$. Также должно быть выполнено $1 \leq x_1, y_1, x_2, y_2 \leq 10^9$.

Потом выведите v — количество вертикальных синих проводов ($0 \leq v$). В следующих v строках выведите по 4 целых числа x_1, y_1, x_2, y_2 — координаты двух точек (x_1, y_1) и (x_2, y_2) , которые соединяет синий очередной провод. Поскольку отрезки вертикальные, должно быть выполнено $x_1 = x_2$. Также должно быть выполнено $1 \leq x_1, y_1, x_2, y_2 \leq 10^9$.

Никакие два отрезка одного цвета не должны иметь общих точек. Множество точек, в которых Паша мог увидеть искру, если бы он построил такую схему, должно совпадать с данным во входных данных множеством точек.

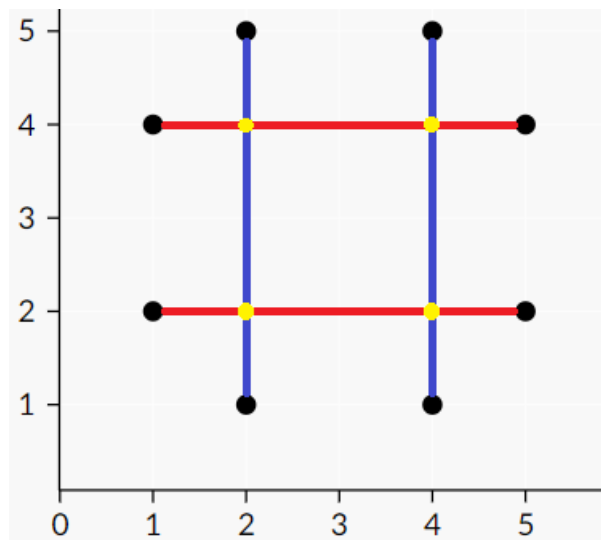
Количество отрезков ($h + v$) должно быть минимально возможным. Можно легко показать, что ответ всегда существует. Если существует несколько возможных ответов, выведите любой.

Примеры

стандартный ввод	стандартный вывод
4 2 2 2 4 4 2 4 4	2 5 2 1 2 1 4 5 4 2 2 1 2 5 4 5 4 1
4 2 1 3 2 2 3 1 2	4 2 1 2 1 3 2 3 2 1 2 1 2 2 3 2 3 3 1 2 1 2 3 2 3 2 2 3 2 1

Замечание

В первом примере Паша мог собрать такую схему:



В этой схеме по 2 провода каждого цвета: красные из (5,2) в (1,2) и из (1,4) в (5,4), синие из (2,1) в (2,5) и из (4,5) в (4,1). Заметим, что он увидит искры ровно в тех точках, которые он записал (обозначены желтым цветом на картинке). Например, искру в точке (2,4) он увидит, так как в этой точке пересекаются второй красный провод и первый синий. Можно доказать, что нужно не меньше 4-х проводов, чтобы получить схему, нужную Паше.

Задача С. Ехаб и странная формула веса

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2.5 секунд
Ограничение по памяти: 256 мегабайт

Дано дерево, состоящее из n вершин. У каждой вершины u есть вес a_u . Гарантируется, что в дереве есть только одна вершина с минимальным весом. У каждой вершины u (кроме вершины с минимальным a_u), есть сосед v , такой, что $a_v < a_u$.

Вам необходимо составить дерево с минимальным весом w , который считается следующим образом:

- Для каждой вершины u , $deg_u \cdot a_u$ прибавляется к w (deg_u обозначает степень вершины u).
- Для каждого ребра $\{u, v\}$, $\lceil \log_2(dist(u, v)) \rceil \cdot \min(a_u, a_v)$ прибавляется к w , где $dist(u, v)$ обозначает количество ребер на пути из u в v в данном дереве.

Формат входных данных

В первой строке записано одно целое число n ($2 \leq n \leq 5 \cdot 10^5$), количество вершин в дереве.

Во второй строке записаны n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$), веса вершин дерева.

Далее следует $n-1$ строка, в каждой из которых записаны по два целых числа u и v ($1 \leq u, v \leq n$). Это означает, что между u и v есть ребро в данном дереве.

Формат выходных данных

Выведите одно целое число: минимальное возможное значение w .

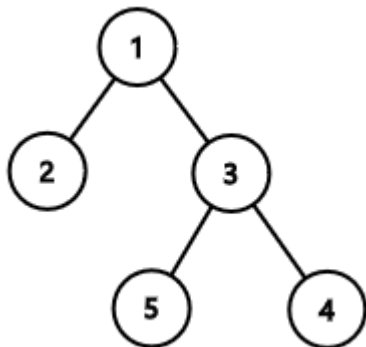
Примеры

стандартный ввод	стандартный вывод
3 1 2 3 1 2 1 3	7
5 4 5 3 7 8 1 2 1 3 3 4 4 5	40

Замечание

В первом примере данное дерево исходно имеет минимальное значение w .

Во втором примере оптимальное дерево имеет следующий вид:



Задача D. Наркоконтроль

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

В маленьком городке «М» начала действовать служба наркоконтроля. Первая задача службы — выяснить, сколько наркоторговцев работает в окрестности города. Агенты службы опросили всех наркозависимых в городе и составили список случаев продажи травки, произошедших за одни сутки, с указанием места и времени наблюдения. Теперь аналитики хотят понять, сколько же на самом деле есть наркоторговцев. Из данных разведки известна максимальная скорость, с которой может двигаться наркоторговец. Аналитики просят вас узнать, какое минимальное количество наркоторговцев могли участвовать во всех зафиксированных случаях продажи травки.

Формат входных данных

На первой строке входного файла содержатся целые числа n и v — количество случаев продажи травки и максимальная скорость наркоторговца ($1 \leq n \leq 100, 1 \leq v \leq 10000$). Следующие n строк содержат описания случаев продажи травки в формате «ЧЧ:ММ x y », где ЧЧ:ММ — время продажи, x и y — координаты места, в котором продавалась травка (для простоты будем считать, что всё происходило на плоскости). Координаты по модулю не превышают 1000. Скорость выражена в км/ч, координаты — в км.

Формат выходных данных

Выведите в выходной файл одно число — минимальное возможное количество наркоторговцев.

Пример

стандартный ввод	стандартный вывод
4 1 12:00 0 0 13:10 0 1 14:00 1 0 15:00 1 1	2

Задача E. Xor the Graph

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дан неориентированный граф, состоящий из N вершин и M рёбер. Каждому ребру присвоено значение, 0 или 1. Операция состоит в том, чтобы выбрать путь (не обязательно простой), и изменить значения, записанные на каждом его ребре на противоположные. Сделайте значения на всех рёбрах нулями за минимальное число операций.

Формат входных данных

На первой строке заданы два целых числа N и M ($1 \leq N \leq 10^5$, $0 \leq M \leq 10^5$).

В каждой из следующих M строк записаны по три целых числа a , b и c . Они означают, что существует ребро между вершинами (a, b) , изначальное значение которого равно c ($1 \leq a, b \leq n$, $0 \leq c \leq 1$).

Формат выходных данных

Выведите количество операций K на первой строке.

В последующих K строках выведите пути, которые вы выбрали:

- Первым числом выведите P — количество вершин в пути
- Затем выведите P целых чисел v_1, v_2, \dots, v_P — индексы вершин. Для всех $1 \leq i \leq P - 1$ (v_i, v_{i+1}) должно являться ребром в заданном графе.

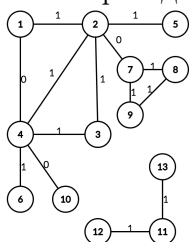
Решение будет считаться верным, если оно использует минимальное количество операций, и при этом $\sum P \leq 4 * M$.

Пример

стандартный ввод	стандартный вывод
13 14	3
1 2 1	5 1 2 3 4 6
2 3 1	8 4 2 7 8 9 7 2 5
3 4 1	3 12 11 13
2 4 1	
1 4 0	
4 6 1	
4 10 0	
2 5 1	
2 7 0	
7 8 1	
8 9 1	
9 7 1	
11 12 1	
11 13 1	

Замечание

Ниже приведён граф в первом тесте.



Задача G. Проблема падишаха

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Мудрый падишах внимательно следит за благополучием своих подданных, когда вершит их судьбы. В частности, на нем все заботы о вступающих в брачный возраст юношах и девушках его страны. И, как положено серьезному правителю, все по науке — перед тем, как творить молодые семьи, падишах провел Глобальное тестирование и по 100-балльной шкале определил совместимость всех юношей и девушек в совместном браке.

А дальше что? Падишах наслышан про задачу о назначении, но ему не нравится ее установка. Действительно, может ли быть спокойна его душа даже в случае всеобщего благополучия, если кому-то из подданных плохо? И можно ли жертвовать интересами хотя бы одной семьи во благо общества? Конечно, нет!

Падишаху милее другая мысль. Он хочет создать максимальное число семей, причем сделать это таким образом, чтобы минимальная совместимость в семье была максимальной. А решить эту неклассическую задачу он просит вас. Помогите падишаху!

Формат входных данных

В первой строке входных данных содержатся два целых числа n и m — количество юношей и количество девушек соответственно ($1 \leq n, m \leq 200$). Последующие n строк содержат по m целых чисел от 0 до 10^9 — коэффициент совместимости соответствующей пары (меньшее значение менее способствует супружеской жизни).

Формат выходных данных

В единственную строку выходного файла выведите наименьший искомый балл, при котором возможно создание максимально возможного количества семейных пар.

Пример

стандартный ввод	стандартный вывод
3 4 77 88 31 67 96 30 2 68 35 39 76 45	76

Задача Н. День рождения

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Митя знаком с m юношами и n девушками и хочет пригласить часть из них на свой день рождения. Ему известно, с какими девушками знаком каждый юноша, и с какими юношами знакома каждая девушка. Он хочет добиться того, чтобы каждый приглашённый был знаком со всеми приглашёнными противоположного пола, пригласив при этом максимально возможное число своих знакомых. Помогите ему это сделать!

Формат входных данных

Входной файл состоит из одного или нескольких наборов входных данных. В первой строке входного файла записано число наборов k ($1 \leq k \leq 20$). В последующих строках записаны сами наборы входных данных.

В первой строке каждого набора задаются числа $0 \leq m \leq 150$ и $0 \leq n \leq 150$. Далее следуют m строк, в каждой из которых записано одно или несколько чисел — номера девушек, с которыми знаком i -й юноша (каждый номер встречается не более одного раза). Строка завершается числом 0.

Формат выходных данных

Для каждого набора выведите четыре строки. В первой из них выведите максимальное число знакомых, которых сможет пригласить Митя. В следующей строке выведите количество юношей и количество девушек в максимальном наборе знакомых. Следующие две строки должны содержать номера приглашённых юношей и приглашённых девушек соответственно. Если максимальных наборов несколько, то выведите любой из них.

Пример

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

Задача I. Декартово

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Государство Иксово состоит из N_x городов, некоторые пары которых связаны дорогами с двусторонним движением. Каждая дорога имеет свою длину. Всего межгородских дорог в стране M_x , причем известно, что из каждого города Иксевщины можно доехать по дорогам до каждого другого города этой страны. Города Иксово пронумерованы натуральными числами от 1 до N_x .

Государство Игреково состоит из N_y городов, некоторые пары которых связаны дорогами с двусторонним движением. Каждая дорога имеет свою длину. Всего межгородских дорог в стране M_y , причем известно, что из каждого города Игреково можно доехать по дорогам до каждого другого города этой страны. Города Игреково пронумерованы натуральными числами от 1 до N_y .

Страна Декартово состоит из $N = N_x \cdot N_y$ городов: каждому городу Декартово во взаимно однозначное соответствие можно поставить пару городов-побратимов (x, y) , где x — город Иксово, а y — город Игреково. Некоторые пары городов Декартово также соединены дорогами с двусторонним движением. Дорог в стране ровно $M = N_x \cdot M_y + N_y \cdot M_x$. При этом дорога между городами (x_1, y_1) и (x_2, y_2) существует только в одном из таких двух случаев:

1. Если $x_1 = x_2$, а между городами y_1 и y_2 Игреково проложена дорога. При этом длина дороги между городами (x, y_1) и (x, y_2) Декартово равно длине дороги между городами y_1 и y_2 Игреково.
2. Если $y_1 = y_2$, а между городами x_1 и x_2 Иксевщины проложена дорога. При этом длина дороги между городами (x_1, y) и (x_2, y) Декартово равно длине дороги между городами x_1 и x_2 Иксево.

Города разных государств между собой дорогами не соединены.

Данная задача состоит из двух подзадач. В обеих подзадачах всю информацию про соединение дорогами задано во входных файлах.

В первой подзадаче требуется определить длину самого короткого пути по дорогам Декартовщины из города $(1, 1)$ в город (N_x, N_y) .

Во второй подзадаче некоторые дороги Декартовщины требуется закрыть. Ваша задача — определить, дороги какой наименьшей суммарной длины можно оставить в Декартовщине, чтобы из любого ее города все еще можно было попасть в любой другой.

Формат входных данных

Первая строка входного файла содержит номер подзадачи, которую требуется решить (1 или 2). Вторая строка содержит натуральные числа N_x и M_x ($1 \leq N_x, M_x \leq 5 \cdot 10^4$) — количество городов и дорог в Иксово. В последующих M_x строках описаны дороги Иксово: в каждой строке по три числа, где первые два задают номера разных городов, соединенных дорогой, а третья есть длиной соответствующей дороги (натуральное число, которое не превышает 10^7).

В следующей строке входного файла указаны натуральные числа N_y и M_y ($1 \leq N_y, M_y \leq 5 \cdot 10^4$) — количество городов и дорог в Игреково. Последующие M_y строк содержат описание дорог Игреково; формат данных и ограничения соответствуют описанным выше.

Формат выходных данных

Выведите единственное целое число — ответ на вопрос подзадачи.

Примеры

стандартный ввод	стандартный вывод
1 3 2 2 1 15 3 1 14 3 2 2 1 15 3 2 15	44
2 3 2 2 1 15 3 1 14 3 2 2 1 15 3 2 15	117

Задача J. Еще одна задача про остов

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Костя с Ваней при подготовке очередной задачи на персистентную двумерную динамическую структуру данных нашли массив a_i размера n , состоящий из целых чисел, каждое из которых находится в промежутке $[0, m - 1]$. Костя, как любитель цветных графов, сразу создал полный неориентированный граф из n вершин, вершине v был задан цвет a_v . Ваня, как любитель взвешенных графов, сразу задал каждому ребру (v, u) вес $f(a_v, a_u)$.

$$f(a, b) = \begin{cases} a + b & a + b < m \\ a + b - m, & \text{иначе} \end{cases}$$

Теперь ребят интересует вопрос: каким же будет вес минимального остовного дерева для нашего графа? Помогите им найти ответ.

Формат входных данных

Программа получает на вход два числа n, m ($1 \leq n \leq 2 \cdot 10^5, 1 \leq m \leq 10^9$).

В следующей строке вводится n целых чисел — массив a_i ($0 \leq a_i \leq m - 1$).

Формат выходных данных

Выведите одно число — суммарный вес остовного дерева.

Пример

стандартный ввод	стандартный вывод
3 2 0 1 1	1

Задача К. Сортировка циклами

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Вам дан массив из n целых положительных чисел a_1, a_2, \dots, a_n . К массиву можно применять следующие операции: выбрать несколько различных индексов этого массива i_1, i_2, \dots, i_k ($1 \leq i_j \leq n$) и переместить число, стоящее на позиции i_1 на позицию i_2 , число стоящее на позиции i_2 на позицию i_3, \dots , число на позиции i_k на позицию i_1 . То есть сдвинуть элементы по циклу $i_1 \rightarrow i_2 \rightarrow \dots \rightarrow i_k \rightarrow i_1$.

К примеру, если у вас есть $n = 4$, массив $a_1 = 10, a_2 = 20, a_3 = 30, a_4 = 40$, и вы выбрали три индекса $i_1 = 2, i_2 = 1, i_3 = 4$, то получившийся массив будет $a_1 = 20, a_2 = 40, a_3 = 30, a_4 = 10$.

Вам дано целое неотрицательное число s . Отсортируйте массив с помощью минимально возможного количества таких операций, при условии, что сумма размеров циклов для всех операций не превосходит s , или скажите, что это невозможно.

Формат входных данных

Первая строка ввода содержит числа n и s ($1 \leq n \leq 200\,000, 0 \leq s \leq 200\,000$) — число элементов массива и верхнюю границу на сумму длин циклов. Следующая строка содержит n целых чисел a_1, a_2, \dots, a_n — элементы массива ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Если невозможно упорядочить массив, используя циклы с суммарной длиной меньше либо равной s , выведите единственное число «-1».

В противном случае, в первой строке выведите единственное число q — минимальное количество операций, требующееся для того, чтобы упорядочить массив.

На следующих $2 \cdot q$ строках выведите описания операций в том порядке, в каком их нужно применять к массиву. Описание i -й операции начинается со строки с единственным числом k ($1 \leq k \leq n$) — длины цикла (т.е. количества выбранных индексов). Следующая строка должна содержать k различных целых чисел i_1, i_2, \dots, i_k ($1 \leq i_j \leq n$) — сами индексы.

Суммарная длина циклов должна быть меньше или равна s , и массив должен быть упорядочен после применения всех q операций. Если есть несколько возможных ответов с оптимальным q , выведите любой из них.

Примеры

стандартный ввод	стандартный вывод
5 5 3 2 3 1 1	1 5 1 4 2 3 5
4 3 2 1 4 3	-1
2 0 2 2	0

Замечание

В первом примере также возможно упорядочить массив двумя операциями суммарной длины 5: сначала применить цикл $1 \rightarrow 4 \rightarrow 1$ (длины 2), потом применить цикл $2 \rightarrow 3 \rightarrow 5 \rightarrow 2$ (длины 3). Однако, это неверный ответ, поскольку в задаче требуется минимизировать количество операций, а минимальное число операций для этого теста равно единице.

Во втором примере, можно упорядочить массив двумя циклами общей длины 4 $1 \rightarrow 2 \rightarrow 1$ и $3 \rightarrow 4 \rightarrow 3$. Но невозможно циклами длиной не больше 3.

В третьем примере, массив уже упорядочен, поэтому операции не нужны. Общая длина пустого множества циклов принимается равной нулю.