

Задача А. Декомпозиция

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.6 секунд
Ограничение по памяти: 256 мегабайт

Рассмотрим дерево T . Назовем деревом декомпозиции корневое дерево $D(T)$. Выберем любую из вершин дерева T , назовем ее r . Рассмотрим все компоненты связности дерева T , после удаления вершины r : S_1, S_2, \dots, S_k . Тогда корнем $D(T)$ будет вершина r , а детьми r в $D(T)$ будут $D(S_1), D(S_2), \dots, D(S_k)$.

Вам дано дерево T . Найдите дерево декомпозиции высоты не более 20. Высота дерева — максимальное число вершин в пути от корня до какой-то вершины.

Формат входных данных

Первая строка содержит n ($1 \leq n \leq 2 \cdot 10^5$) — количество вершин дерева.

Следующие $n - 1$ строк содержат пары чисел u_i, v_i ($1 \leq u_i, v_i \leq n$), описывающие рёбра дерева.

Формат выходных данных

Выведите n чисел, где i -е — родитель вершины i в дереве декомпозиции. Если вершина — корень, выведите 0.

Примеры

стандартный ввод	стандартный вывод
3 1 2 2 3	2 0 2
9 3 2 4 2 1 2 5 1 1 6 7 6 6 8 8 9	0 1 2 2 1 1 6 6 8

Задача В. Красим дерево

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.6 секунд
Ограничение по памяти:	256 мегабайт

Дано взвешенное дерево. Вам необходимо выполнять 2 типа запросов:

- «1 v d c » — покрасить все вершины на расстоянии не более d от v в цвет c . Изначально все вершины имеют цвет 0.
- «2 v » — вывести цвет вершины v .

Формат входных данных

Первая строка содержит целое число n ($1 \leq n \leq 10^5$) — количество вершин в дереве.

Следующие $n - 1$ содержат тройки чисел u_i, v_i, w_i ($1 \leq u_i, v_i \leq n$, $1 \leq w_i \leq 10^4$). i -е ребро соединяет вершины u_i, v_i и имеет вес w_i .

В следующей строке содержится количество запросов q ($1 \leq q \leq 10^5$).

Каждая из следующих q строк содержит запрос какого-то типа:

- 1 v d c ($1 \leq v \leq n$, $0 \leq d \leq 10^9$, $0 \leq c \leq 10^9$).
- 2 v ($1 \leq v \leq n$).

Формат выходных данных

Для каждого запроса второго типа выведите ответ на него.

Примеры

стандартный ввод	стандартный вывод
2 1 2 1 4 1 1 1 10 1 1 0 20 2 1 2 2	20 10
5 1 2 30 1 3 50 3 4 70 3 5 60 8 1 3 72 6 2 5 1 4 60 5 2 3 2 2 1 2 144 7 2 4 2 5	6 6 0 5 7

Задача С. Гоша и праздники

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	512 мегабайт

Как известно, жители планеты Иннополис — очень педантичные люди. И даже когда дело касается праздников, они всегда хотят быть уверенными в том, что все пройдёт как по маслу. Так, расписание празднований всех событий на этой планете составлено почти на три миллиона лет вперёд! Гоша — большой любитель праздников. Он решил прилететь в какой-то из городов планеты Иннополис и посетить как можно больше праздников.

На планете Иннополис n городов, соединённых $n - 1$ двунаправленными дорогами так, что из любого города планеты можно добраться до любого другого, возможно, посещая другие города. Каждое событие на Иннополисе характеризуется номером города c_i , в котором оно будет отпраздновано, и номером дня d_i , в который его будут праздновать.

Гоша настолько везучий человек, что день его прибытия на планету имеет номер 0 в календаре планеты Иннополис, причём исходно он может прилететь в любой город планеты. Гоша решил узнать, какое максимальное количество праздников он может посетить на этой планете. Для этого он обратился за помощью к вам.

Формат входных данных

В первой строке входного файла задано одно число n ($n \geq 1$) — количество городов Иннополиса.

В следующих $n - 1$ строках заданы описания дорог, каждая дорога задается числами a_i , b_i и l_i ($1 \leq a_i, b_i \leq n$; $l_i \geq 1$) — номера городов, которые соединяет дорога и число дней, необходимых на ее преодоление.

В следующей строке задано число m ($m \geq 1$) — число праздников на планете.

В следующих m строках заданы пары чисел c_i и d_i ($1 \leq c_i \leq n$; $d_i \geq 1$) — номер города и номер дня, в который пройдёт i -й праздник.

Ограничения: $n \leq 2 \cdot 10^5$, $m \leq 2 \cdot 10^5$, $l_i \leq 10^9$, $d_i \leq 10^9$.

Формат выходных данных

В единственной строке выходного файла выведите одно число — максимальное количество праздников, которое может посетить Гоша.

Примеры

стандартный ввод	стандартный вывод
4 1 2 1 2 3 1 2 4 3 4 1 3 2 4 3 1 4 5	3
11 2 1 2 3 2 5 4 1 5 5 2 4 6 5 1 7 1 2 8 3 4 9 6 2 10 7 2 11 2 2 9 1 67 1 34 11 16 5 97 4 70 2 20 2 61 2 26 2 70	8
10 2 1 1 3 2 4 4 2 4 5 3 2 6 4 5 7 5 4 8 3 1 9 6 2 10 7 5 9 7 34 10 82 2 48 3 66 8 98 2 66 3 3 8 59 5 22	8

Задача D. Гений Евгенийевич

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

Гений Евгенийевич давно мечтал о дереве. Сейчас в моде деревья с n вершинами, на каждой из которых написано число x_i . Долгожданный день настал, и Гений Евгенийевич смог себе приобрести такое дерево. Вернувшись из магазина после покупки, он начал считать различные величины на этом дереве. Все ему давалось очень просто, пока он не решил найти путь в этом дереве, значение величины $l(p(u, v)) \cdot \min(x_u, x_v)$ для которого максимально. Здесь $p(u, v)$ — путь между вершинами u и v , $l(p)$ — длина пути p в ребрах.

Уже который день Гений Евгенийевич не выходит на улицу. Его друзья, естественно, начали волноваться. Они просят Вас помочь Гению Евгенийевичу справиться с задачей.

Формат входных данных

В первой строке входных данных дано число n ($1 \leq n \leq 10^5$) — количество вершин в дереве.

Во второй строке даны n чисел x_i ($1 \leq x_i \leq 10^9$) — числа написанные на вершинах. На i -й вершине написано число x_i .

В следующих $n - 1$ -й строке идут пары чисел a_i, b_i ($1 \leq a_i, b_i \leq n$) — рёбра в дереве.

Формат выходных данных

Выведите максимальное значение величины, описанной в условии.

Пример

стандартный ввод	стандартный вывод
8 9 2 8 8 7 2 1 2 2 1 3 1 4 1 5 2 6 4 7 5 8 5	21

Задача E. Найти ближайшую

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

Дано дерево из n вершин, цвет i -й вершины равен a_i . Необходимо обработать q запросов (v_i, c_i) : найти расстояние от v_i до ближайшей вершины цвета c_i . Расстояние между вершинами — минимальное количество рёбер в пути между ними.

Формат входных данных

Первая строка содержит n ($1 \leq n \leq 10^5$).

Следующая строка содержит $n - 1$ число p_1, \dots, p_{n-1} ($0 \leq p_i < i$). p_i — отец вершины i .

Следующая строка содержит числа a_1, \dots, a_n ($0 \leq a_i < n$).

Следующая строка содержит число q ($1 \leq q \leq 10^5$).

Следующие q строк содержат числа v_i, c_i ($0 \leq v_i < n, 0 \leq c_i < n$).

Формат выходных данных

Для каждого запроса выведите расстояние до ближайшей вершины требуемого цвета, или -1 , если такой нет.

Пример

стандартный ввод	стандартный вывод
5	0 1 2 -1 2 1 2 1 1
0 1 1 3	
1 2 3 2 1	
9	
0 1	
0 2	
0 3	
1 0	
2 1	
2 2	
3 3	
3 1	
4 2	

Задача F. Интерактивная вершина

Имя входного файла:	<i>стандартный ввод</i>
Имя выходного файла:	<i>стандартный вывод</i>
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Это интерактивная задача

У Ильдара есть дерево из n вершин и он показал его вам. Он выбирает одну вершину u как специальную вершину, но он не сообщает вам ничего о ней!

Вместо этого, вы можете задавать ему вопросы. Для каждого вопроса вы должны выбрать вершину x , натуральное число k и k вершин v_1, v_2, \dots, v_k и он вам скажет, правда ли, что $\min(\text{dist}(u, v_i)) \geq \text{dist}(u, x)$. Здесь, $\text{dist}(p, q)$ это количество ребер на простом пути между вершинами p и q в дереве.

Вы должны угадать специальную вершину за не больше, чем $4 \lceil \log_2 n \rceil$ вопросов.

Ильдар очень добрый, поэтому он не будет менять специальную вершину между вашими вопросами (другими словами, интерактор неадаптивный).

Поскольку ограничения большие и flush это тяжелая операция, убедитесь, что вы не делаете операцию flush слишком часто. Рекомендуется делать flush только после вывода каждого вопроса.

Протокол взаимодействия

Процесс взаимодействия начинается с того, что на первой строке вводится целое число n : количество вершин в дереве Ильдара ($2 \leq n \leq 200\,000$).

Каждая из следующих $n - 1$ строк содержит два целых числа u и v ($1 \leq u, v \leq n$), означающих ребро между u и v . Гарантируется, что данные ребра образуют дерево.

После этого, вы можете задавать вопросы.

Чтобы задать вопрос, выведите одну строку, содержащую “? k ” ($1 \leq k \leq n$), целое число x ($1 \leq x \leq n$) и затем k различных целых чисел v_1, v_2, \dots, v_k ($1 \leq v_i \leq n$). Разделяйте соседние числа в строке ровно одним пробелом. Затем сделайте flush выходного потока.

После каждого вопроса, считайте одно целое число $ans \in \{0, 1\}$. Если $\min(\text{dist}(u, v_i)) \geq \text{dist}(u, x)$, тогда ans будет равно 1. Иначе, ans будет равно 0.

Когда вы нашли специальную вершину u ($1 \leq u \leq n$), выведите одно целое число “! u ”, сделайте flush выходного потока и завершите работу программы.

Ваше решение получит Wrong Answer или Time Limit Exceeded если вы сделаете больше чем $4 \lceil \log_2 n \rceil$ вопросов.

Ваше решение получит Idleness Limit Exceeded если оно не совершает никаких действий или не делает flush выходного потока.

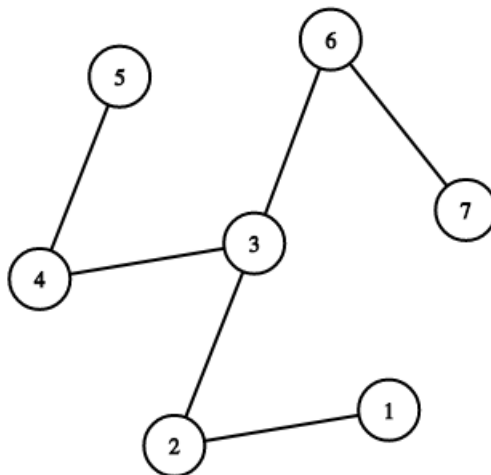
Чтобы сделать flush выходного потока, вы можете использовать:

- `fflush(stdout)` или `cout.flush()` в C++;
- `System.out.flush()` в Java;
- `flush(output)` в Pascal;
- `stdout.flush()` в Python;
- используйте документацию других языков.

Примеры

<i>стандартный ввод</i>	<i>стандартный вывод</i>
5 1 2 1 3 1 4 1 5 1	? 4 1 2 3 4 5 ! 1
5 1 2 1 3 1 4 1 5 0 0 0 0	? 4 1 2 3 4 5 ? 3 1 2 3 4 ? 2 1 2 3 ? 1 1 2 ! 2
7 1 2 2 3 3 4 4 5 3 6 6 7 1	? 3 3 5 7 1 ! 3

Замечание



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 N. 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. 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 J. 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 К. Двойная сортировка II

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

Даны две перестановки a и b , обе размера n . Перестановка размера n — это массив из n элементов, в который каждое целое число от 1 до n входит ровно один раз. Элементы в каждой перестановке пронумерованы от 1 до n .

Вы можете выполнять следующую операцию любое количество раз:

- выбрать целое число i от 1 до n ;
- обозначим за x такое целое число, что $a_x = i$. Поменять местами a_i и a_x ;
- обозначим за y такое целое число, что $b_y = i$. Поменять местами b_i и b_y .

Вы должны отсортировать обе перестановки **в порядке возрастания** (то есть должны выполняться условия $a_1 < a_2 < \dots < a_n$ и $b_1 < b_2 < \dots < b_n$) за **минимальное количество операций**. Обратите внимание, что обе перестановки должны быть отсортированы после того, как вы выполните выбранную вами последовательность операций.

Формат входных данных

В первой строке задано одно целое число n ($2 \leq n \leq 3000$).

Во второй строке заданы n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$; все a_i различны).

В третьей строке заданы n целых чисел b_1, b_2, \dots, b_n ($1 \leq b_i \leq n$; все b_i различны).

Формат выходных данных

Сначала выведите одно целое число k ($0 \leq k \leq 2n$) — минимальное количество операций, за которое можно отсортировать обе перестановки. Обратите внимание: можно показать, что $2n$ операций всегда достаточно.

Затем выведите k целых чисел op_1, op_2, \dots, op_k ($1 \leq op_j \leq n$), где op_j — значение i , выбранное для j -й операции.

Если ответов несколько, выведите любой из них.

Примеры

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

Задача L. 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.

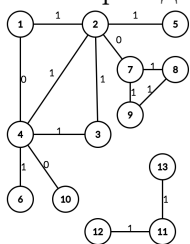
Решение будет считаться верным, если оно использует минимальное количество операций, и при этом $\sum P \leq 4 * M$.

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 M. 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 , ..., 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.