

Задача А. Стрельба из пушки

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

Галактическая Федерация хочет разрушить дом Рика при помощи новейшего планетарного уничтожителя NX5. Он представляет из себя пушку, расположенную на земле в точке $(0, 0)$, из которой можно стрелять энергетическим пучком под любым углом относительно оси X на расстояние k .

Если по пути пучок касается верха какого-то препятствия, то это препятствие уничтожается. Если же пучок сталкивается с ним, его вся энергия пучка уходит в это препятствие, уничтожает его, и пучок прекращает свое движение. Если, пролетев расстояние k , пучок ни с чем не сталкивается, то вся его энергия уходит вниз перпендикулярно поверхности земли (как молния), уничтожая препятствие на пути к земле, если оно есть. Также NX5 может выстрелить параллельно оси X , и пучок будет лететь, пока не пролетит расстояние k или не наткнется на препятствие.

На расстоянии d от устройства стоит энергетический щит высоты w , который является препятствием для пучка. На расстоянии $2d$ от земли находится дом семьи Смитов, который имеет высоту h . Найдите минимальный угол, под которым враги могут выстрелить из уничтожителя и разрушить дом, или выведите -1 , если такого угла не существует.

Формат входных данных

В одной строке и единственной строке входных данных через пробел дано четыре целых числа d, h, w, k ($1 \leq d, h, k \leq 10^9$; $0 \leq w \leq 10^9$) — расстояние от пушки до щита по координате X , высота дома, высота электрического щита и максимальная дальность полета пучка.

Формат выходных данных

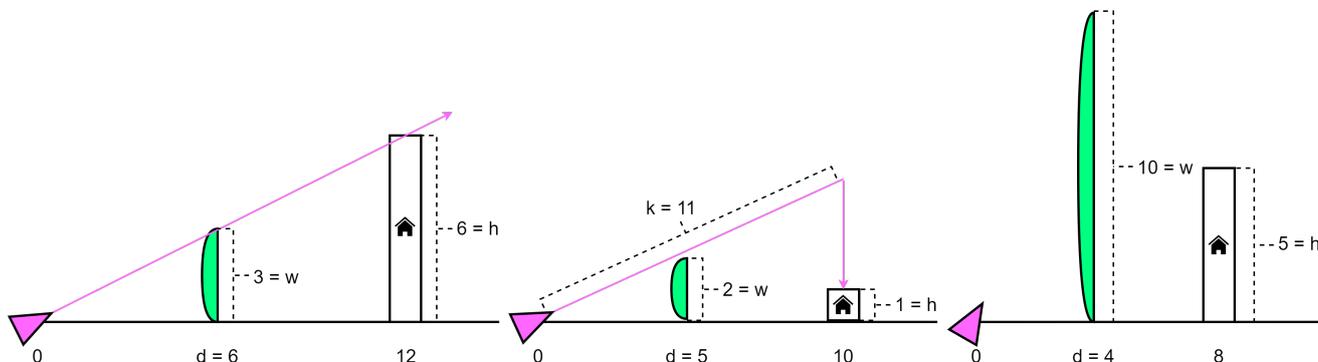
Выведите минимальный угол, под которым нужно стрелять, чтобы разрушить дом героев (в градусах с точностью до четырех знаков после запятой), или -1 , если это невозможно.

Примеры

стандартный ввод	стандартный вывод
6 6 3 20	26.5651
5 1 2 11	24.62
4 5 10 12	-1

Замечание

Ниже приведены иллюстрации к тестовым примерам.



Задача В. Иерархия цитадели

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

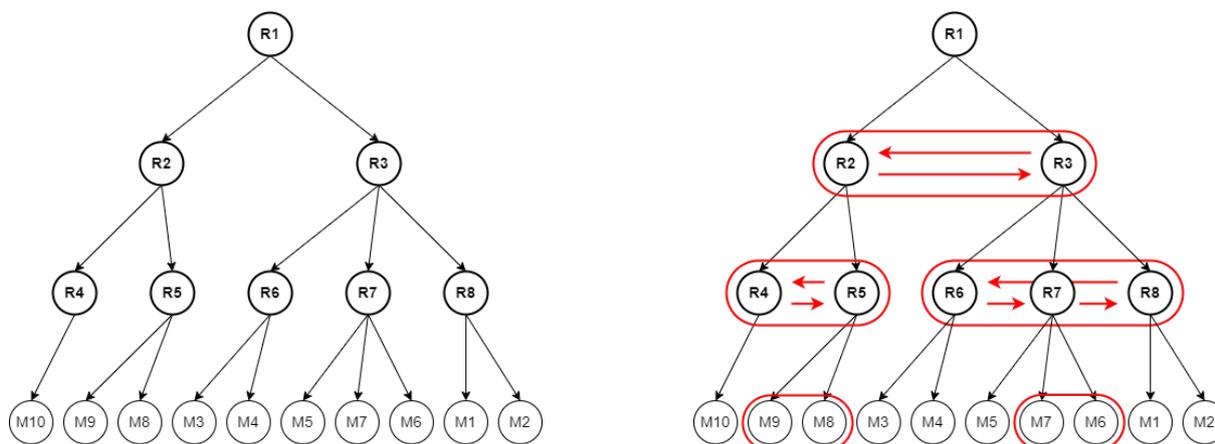
Как известно, в Цитадели Риков обитает бесчисленное множество Риков и Морти (а именно — n Риков и m Морти). Чтобы в новой Цитадели не было полного хаоса, было решено построить четкую иерархию, благодаря которой всегда можно будет быстро определить ~~какой Морти~~ кто виноват в том или ином происшествии.

Для начала было решено пронумеровать всех Риков по уменьшению важности от 1 до n , а Морти — по увеличению неважности от 1 до m . Иерархию обитателей цитадели решили изобразить в виде подвешенного дерева, при чем

- в дереве ровно n внутренних вершин и m листьев, и все листья дерева находятся на одной глубине;
- все внутренние вершины заняты Риками, а все листья заняты Морти (разумеется, все Морти должны находиться в самом низу иерархии);
- номера всех Риков на любом уровне меньше, чем номера Риков на следующих уровнях (в частности, в корне дерева всегда находится Верховный Рик под номером 1);
- на каждом уровне Рики пронумерованы по возрастанию слева-направо;
- у каждого Рика **до предпоследнего слоя** есть хотя бы два непосредственных подчиненных.

Рики, разумеется, быстро справились с построением такой иерархии, а вот Морти в панике выстроились на нижнем уровне в каком-то случайном порядке. Посмотрев на этот хаос, Рики решили, что раз они все равно не очень любят правила, то правило про упорядоченность Риков на каждом уровне можно отменить, а вот Морти на нижнем уровне надо упорядочить по возрастанию номеров слева-направо.

Для этого каждому Рику было разрешено поменять своих непосредственных подчиненных местами произвольным образом. При этом менять множество своих подчиненных, то есть брать новых или отдавать старых кому-то еще, а также перемещаться на другой уровень иерархии, запрещено. Получится ли у Риков таким образом упорядочить всех Морти по возрастанию? Ниже приведен пример возможного решения:



Изначальная иерархия и действия, необходимые для упорядочивания Морти

Формат входных данных

В первой строке через пробел перечислены два целых числа n и m — количество Риков и Морти в Цитадели, соответственно ($2 \leq n, m \leq 10^5$).

Во второй строке через пробел перечислены m различных целых чисел a_i — номера Морти в порядке их следования в изначально построенной иерархии слева-направо ($1 \leq a_i \leq m$).

В следующей строке через пробел перечислены числа p_2, \dots, p_n — номера непосредственных начальников Риков с номерами от 2 до n ($1 \leq p_i < i$).

В следующей строке, аналогично, перечислены m целых чисел q_1, \dots, q_m — номера непосредственных начальников всех Морти, в порядке, в котором они следуют в исходной иерархии ($1 \leq q_i \leq n$). **Обратите внимание**, что q_1 — номер начальника Морти с номером a_1 , а не с номером 1.

Гарантируется, что структура иерархии соответствует заданным в условии ограничениям.

Формат выходных данных

В единственной строке выведите «YES» (без кавычек), если Рики, меняя местами непосредственных подчиненных, могут упорядочить всех Морти по возрастанию номеров, и «NO» иначе.

Примеры

стандартный ввод	стандартный вывод
4 4 1 3 2 4 1 1 1 2 2 3 4	NO
8 10 10 9 8 3 4 5 7 6 1 2 1 1 2 2 3 3 3 4 5 5 6 6 7 7 7 8 8	YES

Задача С. Защитное поле

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	10 секунд
Ограничение по памяти:	256 мегабайт

Рик и Морти ночуют на планете, на которой конкретно этой ночью температура достигнет абсолютного нуля. Оставшись без средств передвижения и без порталной пушки, они используют для выживания все, что могут, однако основной проблемой остается поиск пропитания.

К счастью, неподалеку от них находятся n особей довольно вкусных обитателей данной планеты; i -я особь находится в точке с координатами (x_i, y_i) , если считать местность вокруг Рика и Морти приблизительно плоской. К сожалению, за ночь при абсолютном нуле эти существа замерзают на следующие несколько лет, поэтому Рик решил позаботиться о том, чтобы уберечь некоторую часть из них от холода и обеспечить себе и Морти питание на следующие дни. Для этого у Рика есть генератор защитного поля, которое не позволит температуре в некоторой круглой области опуститься так низко.

Центр защитного поля Рик может выбрать сам, но чтобы не тратить энергию попусту, он хочет установить защитное поле как можно меньшего радиуса. Однако Морти просит Рика уместить в защитном поле хотя бы $\lceil \frac{n}{2} \rceil$ особей, чтобы спасти их от вынужденной многолетней спячки.

Помогите Рикю найти минимальный радиус окружности, внутри которой содержится хотя бы половина существ-обитателей этой планеты.

Формат входных данных

В первой строке входных данных дано целое число n — количество съедобных особей на планете ($1 \leq n \leq 400$).

В следующих n строках даны n пар вещественных чисел x_i и y_i — координаты местоположения каждой особи ($-10^9 \leq x_i, y_i \leq 10^9$). Существа ленивые и никуда не двигаются, поэтому их положения постоянны.

Формат выходных данных

Выведите три числа — центр окружности защитного поля минимального радиуса, покрывающего хотя бы половину существ, и сам этот радиус. Все числа необходимо вывести с погрешностью не более 10^{-6} .

Примеры

стандартный ввод	стандартный вывод
8 4 3 3 4 -3 4 -4 3 -4 -3 -3 -4 3 -4 4 -3	0 3 4
4 1.5 1.5 1.5 -1.5 -1.5 -1.5 -1.5 1.5	1.5 0 1.5

Задача D. Симметричные карты

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

Как это изредка бывает, вся семья Смитов собралась вместе за столом после ужина, чтобы сыграть в карточную игру. В этой игре есть карточки, на каждой из которых написана какая-то цифра от 0 до 9.

В какой-то момент (почти сразу) Рик сказал, что ему скучно, и ушел, забрав с собой Морти, и играть остались только Бет, Джерри и Саммер. Для очередной игры они решили каждый построить последовательность из a , b и c карточек, соответственно (назовем эти последовательности A , B и C). Но чтобы игра получилась интересной, обязательно должны выполняться следующие условия:

- если выложить подряд последовательности A и B (именно в таком порядке) вместе, получившаяся последовательность должна читаться одинаково слева-направо и справа-налево;
- аналогично для последовательностей A и C , их конкатенация должна быть палиндромом;
- и аналогично для последовательностей B и C .

Чтобы не перестраивать много раз последовательности в поисках лучшего варианта, Смиты решили попросить вас общее количество последовательностей из карточек, удовлетворяющих условию, ведь самый умный в семье, Рик, явно не захочет этим заниматься.

Поскольку это число может быть очень большим, найдите его остаток по модулю числа $10^9 + 7$.

Формат входных данных

В первой и единственной строке ввода через пробел перечислены три целых числа a , b и c — длины последовательностей, которые хотят получить Бет, Джерри и Саммер, соответственно ($1 \leq a, b, c \leq 10^6$).

Формат выходных данных

В единственной строке выведите целое число — количество возможных троек последовательностей карточек с длинами a , b , c , удовлетворяющих условию, по модулю $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
2 3 4	100
101 102 103	193000119

Задача Е. Эффективный двигатель

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

Как известно, тарелка Рика летает, используя энергию, вырабатываемую обитателями карманной вселенной, созданной самим Риком. После прошлого инцидента Рик долго размышлял, какие меры предосторожности принять, чтобы снова не оказаться без возможности завести летающую тарелку в самый неподходящий момент, и наконец придумал следующее решение.

Рик решил, что теперь под капотом тарелки будет находиться не одна карманная вселенная, а целых $n!$ При чем, чтобы жители этих карманных вселенных не успевали развивать свои цивилизации до такого уровня, на котором они смогут сами изобрести свои карманные вселенные, некоторые из них Рик периодически будет перезапускать.

Изначально все n карманных вселенных находятся в замороженном состоянии и не функционируют. Механизм перезапуска устроен следующим образом: перед i -м полетом летающей тарелки Рик по очереди пройдет по всем вселенным, номера которых делятся на i (включая саму i -ю), и поменяет их состояние:

- если соответствующая вселенная была заморожена или была раньше перезапущена, Рик спустится в нее и представит ее обитателям технологию, которая позволит им вырабатывать для него энергию;
- если же вселенная уже функционировала, Рик отправляет ее на перезапуск, откатывая развитие цивилизации в ней на стартовую ступень.

Если в какой-то момент перед полетом оказывается, что ни одна вселенная не готова к выработке энергии, на этот случай у Рика есть одна запасная, так что беспокоиться не о чем.

Теперь Рика интересует, сколько вселенных будут функционировать после первых n полетов летающей тарелки. Помогите ему посчитать это количество. Полеты нумеруются от 1 до n , как и карманные вселенные.

Формат входных данных

В единственной строке дано число n — количество карманных вселенных, установленных в летающую тарелку ($1 \leq n \leq 10^{18}$).

Формат выходных данных

В единственной строке выведите ответ на задачу.

Пример

стандартный ввод	стандартный вывод
2	1

Задача F. Артефакты

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	512 мегабайт

Рик и Морти обнаружили в другом измерении карту планеты, на которой изображены n пунктов раскопок, соединенных тропинками. Тропинка с номером i соединяет пункты с номерами u_i и v_i и имеет длину c_i .

Разумеется, Рик сразу заметил, что количество тропинок равняется в точности $n - 1$, и из любого пункта можно добраться до любого другого. Иными словами, структура дорог и пунктов представляет из себя дерево, но для Морти это определение слишком сложное, поэтому Рик оставил Морти изучать теорию графов, а сам отправился исследовать это измерение.

Инопланетный информатор сообщил ему, что всего существует k видов артефактов, и в пункте номер i хранится артефакт вида a_i . Так очень удачно совпало, что у Рика очередное соревнование с одним из известных расхитителей космических гробниц, и для победы Рик нужно собрать все k различных видов артефактов.

Одной из проблем является то, что внутри этого измерения не работают никакие продвинутые технологии. Поэтому Рик может заранее создать порталы в запланированных стартовом и конечном пункте маршрута, а вот остальной маршрут придется пройти пешком. Чтобы сэкономить свое время, Рик хочет заранее выбрать стартовый пункт, конечный пункт и сам путь (не обязательно простой) так, чтобы пройденное им расстояние было минимальным, и при этом на пути он бы собрал все различные виды артефактов.

Рик, конечно, и сам может справиться с поиском такого кратчайшего пути, но, может быть, у вас есть время заняться этим, пока он собирает всю необходимую для путешествия экипировку?

Формат входных данных

В первой строке через пробел даны два целых числа n и k ($2 \leq n \leq 10^5$; $1 \leq k \leq 6$) — количество пунктов и необходимое количество артефактов.

В следующей строке через пробел даны n целых чисел a_i — виды артефактов в каждом пункте ($0 \leq a_i \leq k$). В случае, если $a_i = 0$, считается, что в вершине не хранится никакой из видов артефактов.

В следующих $n - 1$ строках даны тройки целых чисел u_i, v_i, c_i , обозначающие наличие тропинки длины c_i между пунктами u_i и v_i ($1 \leq u_i, v_i \leq n$; $1 \leq c_i \leq 10^9$). Гарантируется, что структура графа представляет из себя дерево.

Формат выходных данных

В случае, если невозможно собрать k различных видов артефактов, выведите «-1» (без кавычек), иначе сообщите **минимальное** расстояние, которое придется пройти, чтобы собрать все виды артефактов.

Примеры

стандартный ввод	стандартный вывод
5 4 1 3 2 4 4 1 3 1 2 3 1 3 4 1 4 5 1	4
5 5 1 3 2 4 4 1 3 1000 2 3 5123 3 4 3341 4 5 7197	-1
4 3 0 1 2 3 1 2 10 2 3 1 3 4 50	51

Замечание

В первом примере одним из оптимальных путей будет $1 \rightarrow 3 \rightarrow 2 \rightarrow 3 \rightarrow 4$.

В втором примере у нас нет пункта, в котором находится артефакт под номером 5, а значит невозможно собрать все пять артефактов.

В третьем примере одним из оптимальных путей будет $2 \rightarrow 3 \rightarrow 4$.

Задача G. Незваные гости

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

Рик решил, что пока он работает над довольно серьезным изобретением в своей новой лаборатории, он хочет знать, кто за это время посещает Землю, ведь среди таких посетителей могут оказаться как люди из Галактической Федерации, так и более опасные личности.

Для этого Рик классифицировал всех возможных существ во Вселенной и разбил их на n групп по степени их опасности или подозрительности.

Система логирования устроена довольно просто, и в тот момент, когда кто-то с категорией опасности i прилетает за Землю, она делает запись $(i +)$, а когда кто-то с такой категорией опасности покидает планету — делает запись $(i -)$. Известно, что в момент запуска системы на планете находятся только люди, которых Рик вообще не воспринимает как угрозу, и поэтому не отнес ни к одной категории.

К сожалению, Морти опять нажал не на тот переключатель на стене, и в системе логирования все сбилось — она все еще исправно делает записи, однако эти записи могут быть перепутаны и следовать в произвольном порядке.

В какой-то момент Рик посмотрел на логи, в которых уже накопилось m записей, и обеспокоился тем, что из некоторых подозрительных категорий планету могло посещать достаточно большое количество личностей. По записям в логах помогите Рикку определить минимальное и максимальное возможное число различных ~~людей~~ существ из каждой категории, которые посещали Землю. Разумеется, никто не может прилететь на Землю два раза подряд, предварительно не улетев перед этим.

Формат входных данных

В первой строке ввода через пробел даны два целых числа n и m — количество категорий существ и количество записей в логах ($1 \leq n \leq 10^5$; $1 \leq m \leq 3 \cdot 10^5$).

В следующих m строках даны записи логирующей системы. В одной записи содержится число x_i и символ '+' — кто-то из x_i -й категории прилетел на Землю, или '-' — кто-то покинул планету ($1 \leq x_i \leq n$).

Формат выходных данных

В первой строке выведите n целых чисел через пробел, i -е число должно быть равно минимально возможному количеству различных посетителей с i -й категорией опасности.

Во второй строке в том же формате выведите максимально возможные количества посетителей каждой категории.

Примеры

стандартный ввод	стандартный вывод
2 4 1 + 1 + 2 + 1 -	1 1 2 1
3 10 1 + 1 - 1 + 2 - 2 + 2 - 1 - 2 + 3 - 3 +	1 1 1 2 2 1

Задача Н. Портальная пушка

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	1024 мегабайта

Рик прекрасно знает, что Морти нельзя доверять портальную пушку, но в последнее время Морти так надоел ему просьбами подарить ему собственную портальную пушку, что Рик сдался и сделал ему мини-версию.

Однако, чтобы Морти при этом как-то развивался (или просто чтобы ему было сложнее ее использовать), Рик сделал ее устройство достаточно запутанным. Всего на портальной пушке Морти есть n параметров, каждый из которых может принимать значения от 'a' до 'z'. Обозначим параметр под номером i за p_i .

За одно действие Морти может:

- выбрать произвольный номер параметра i от 1 до n и изменить его значение на произвольное другое c от 'a' до 'z';
- выбрать два возможных значения c_1 и c_2 от 'a' до 'z' и заменить значение **у всех** параметров, текущее значение которых равно c_1 , на c_2 ;
- проверить, можно ли создать порталы в локациях с идентификаторами (l_1, r_1) и (l_2, r_2) (где $l_1 \leq r_1$ и $l_2 \leq r_2$) — для этого набор параметров на позициях $[l_1, r_1]$ должен поэлементно совпадать с набором параметров на позициях $[l_2, r_2]$, то есть должно выполняться

$$\begin{cases} r_1 - l_1 = r_2 - l_2 \\ p_{l_1+i} = p_{l_2+i} \end{cases} \quad \text{для всех } 0 \leq i \leq r_1 - l_1.$$

Поскольку Морти не шибко умный, ему сложно быстро проверять наборы параметров на равенство, а случайно сломать портальную пушку не хотелось бы. Помогите ему для каждого действия третьего типа понять, может ли он создать порталы в желаемых локациях, или нет.

Формат входных данных

В первой строке ввода дана строка p длины n , состоящая из маленьких латинских букв — начальные значения параметров пушки ($1 \leq n \leq 2 \cdot 10^5$).

В следующей строке дано единственное целое число q — количество действий, которые совершает Морти ($1 \leq q \leq 2 \cdot 10^5$).

В следующих q строках перечислены описания действий:

- «1 i c » — присвоить i -му параметру значение c ($1 \leq i \leq n$; 'a' $\leq c \leq$ 'z');
- «2 c_1 c_2 » — заменить все значения всех параметров, равных c_1 , на c_2 ('a' $\leq c_1, c_2 \leq$ 'z');
- «3 l_1 r_1 l_2 r_2 » — проверить возможность создания порталов в локациях (l_1, r_1) и (l_2, r_2) ($1 \leq l_1, r_1, l_2, r_2 \leq n$; $l_1 \leq r_1$; $l_2 \leq r_2$).

Формат выходных данных

На каждое действие третьего типа выведите в отдельной строке «YES» (без кавычек), если Морти может его выполнить, и «NO» иначе.

Примеры

стандартный ввод	стандартный вывод
aaabc 6 3 1 2 2 3 1 4 c 2 c a 3 1 4 2 5 1 3 x 3 1 4 2 5	YES YES NO
abracadabra 7 3 1 4 8 11 1 7 r 2 a c 2 b c 3 1 4 8 11 3 1 4 5 8 3 2 4 6 8	YES YES YES YES

Задача I. Побег из космической тюрьмы

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

В этот раз Рик угодил в тюрьму вместе с Морти. Кажется, они пытались украсть какой-то очень важный кристалл... Но сейчас это уже не важно, нужно помочь им выбраться!

На дверях камеры написан массив a из n целых чисел. Числа в массиве могут повторяться, и известно, что двери тюрьмы открываются тогда и только тогда, когда массив будет отсортирован по неубыванию. Для изменения состояния массива используется специальный прибор, внутри которого спрятана p — перестановка чисел от 1 до n . После одного применения этого прибора, элементы массива меняют позиции в соответствии с этой перестановкой, то есть число a_i перемещается на позицию p_i для каждого i .

Закрывая дверь в камеру, охранник один раз применил этот прибор к исходному отсортированному массиву, после чего с ехидной улыбкой бросил этот прибор Рик — они оба знают, что чтобы вернуть массив в исходное состояние, в худшем случае Рик придется применить этот прибор еще $n! - 1$ раз. Но охранник не знал, что у Морти в кармане завалилась игрушка, способная запоминать и частично восстанавливать состояния объектов.

Рик и Морти составили план: каждую секунду Морти будет запоминать текущее состояние массива, после чего Рик будет применять к массиву перестановку p . Возможность выбраться у них появится тогда, когда для каждого i от 1 до n в игрушке Морти будет запомнено хотя бы одно состояние, в котором на i -й позиции стоит число, равное исходному значению a_i . Тогда Морти сможет по очереди восстановить каждое число в массиве из «правильного» состояния, после чего массив снова будет отсортирован по неубыванию.

Посчитайте, сколько секунд пройдет, прежде чем Рик и Морти смогут выбраться. Считайте, что Рик и Морти настолько сфокусированы на своем плане, что даже если первое применение перестановки p оставит массив отсортированным, они этого не заметят и все равно потратят одну секунду на выполнение одного действия.

Формат входных данных

В первой строке дано единственное целое число n — длина массива на двери и перестановки ($1 \leq n \leq 5 \cdot 10^5$).

Во второй строке через пробел перечислены n целых чисел a_i — изначальное (отсортированное) состояние массива ($1 \leq a_i \leq 10^9$; $a_i \leq a_{i+1}$).

В последней строке через пробел перечислены n различных целых чисел p_i — элементы перестановки, которую совершает одно использование прибора ($1 \leq p_i \leq n$).

Формат выходных данных

Выведите единственное целое число — количество состояний массива, которое будет запомнено в игрушке Морти к моменту, когда заключенные впервые получают возможность с помощью нее выбраться из камеры.

Примеры

стандартный ввод	стандартный вывод
3 1 1 1 3 1 2	1
5 1 1 2 3 3 2 4 3 5 1	3
6 1 1 1 1 2 2 2 3 5 6 4 1	4

Замечание

В третьем примере состояния массива a будут равны

1. 2, 1, 1, 2, 1, 1
2. 1, 2, 1, 1, 1, 2
3. 2, 1, 2, 1, 1, 1
4. 1, 2, 1, 1, 2, 1

Как можно заметить, 2 не появляется на пятом месте до четвертой секунды, а на всех остальных позициях за эти четыре секунды хотя бы раз встретилось нужное число, поэтому ответ равен 4.

Задача J. Халат Рика

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Рик решил постирать свой халат. Но у него не так много времени до очередного безумного приключения, поэтому вместо нормальной стирки он намочил s точек халата специальным раствором и теперь ждет, когда халат высохнет.

Раствор, созданный Риком, обладает подобием разума, поэтому вместо того, чтобы неэффективно испаряться, растекается по халату. Сам же халат можно представить в виде неориентированного графа на n вершинах и m ребрах. Точки, которые Рик намочил раствором — это k различных вершин графа. Также, как, наверное, можно было заметить, Халат Рика очень старый, поэтому в нем имеются t дырок (еще t различных вершин), и как раз через них раствор может стекать.

Рик точно не создал бы раствор, который действует не оптимально, ведь у него не так много времени, поэтому раствор из каждой изначальной точки течет до ближайшей дырки и весь через нее вытекает. На прохождение одного ребра ткани раствору требуется одна единица времени. По одному ребру одновременно может течь любой объем раствора.

Рик посчитал, через какое время весь раствор вытечет из халата через дырки, и считает, что это слишком долго. Поэтому он хочет проделать в халате еще одну дырку в любой вершине, чтобы ускорить процесс. Помогите ему определить, какую вершину стоит продырявить, чтобы минимизировать время, за которое все капли раствора доберутся до ближайших дырок.

Раствор при этом работает моментально, поэтому после нанесения можно дырявить сразу дырки, в которых находится раствор, если это минимизирует ответ. Поскольку возможных выборов конечной вершины может быть несколько, Рик просит вас найти вершину с минимальным номером, при добавлении дырки в которой искомое расстояние минимизировано.

Формат входных данных

В первой строке через пробел даны целые числа n , m , s и t — количество вершин и ребер в халате, а также количество изначально намоченных точек и дырок, соответственно ($1 \leq n, m \leq 2 \cdot 10^5$; $1 \leq s \leq \min(n, 100)$; $1 \leq t \leq n$).

В следующей строке через пробел перечислены k различных целых чисел p_i — номера вершин, которые Рик намочил раствором ($1 \leq p_i \leq n$).

В следующей строке так же перечислены t различных целых чисел q_i — номера вершин, в которых в халате находятся дырки ($1 \leq q_i \leq n$).

Следующие m строк содержат описание ребер графа. В i -й из них через пробел даны два целых числа v_i и u_i , означающие, что в графе есть ребро между вершинами u_i и v_i ($1 \leq v_i, u_i \leq n$).

Формат выходных данных

В единственной строке выведите через пробел номер вершины, которую следует продырявить, и минимальное время высыхания халата.

Разрешается выводить номер вершины, которая уже является дыркой, если ни при каком выборе другой вершины ответ не улучшится. Если возможных ответов несколько, выведите тот из них, в котором номер выбранной вершины минимален (независимо от того, есть в ней уже дырка или нет).

Пример

стандартный ввод	стандартный вывод
7 6 4 1 4 5 6 7 1 1 2 2 3 3 4 4 5 3 6 6 7	3 2