

Задача А. Аккаунты

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

Проникнув на базу группировки «Золотое кольцо», Эггси и его бывший наставник Гарри вскоре получили доступ к ноутбуку Поппи Адамс. Теперь им необходимо вычислить, отыскать, а впоследствии и арестовать всех участников «Золотого кольца». В процессе долгих поисков агенты наткнулись на нужный им документ — список логинов и паролей всех сотрудников, с помощью которых можно войти в их аккаунты во внутренней системе группировки и узнать имя и местоположение каждого преступника.

Но Поппи оказалась очень умна и поэтому перемешала все логины и пароли в списке так, что на первый взгляд он представлен в виде списка случайных строк. Однако ребята из «Кингсман» не менее сообразительны и быстро догадались, как сопоставить две строки из списка так, чтобы одна из них оказалась логином, а вторая — подходящим паролем.

Логин представляет из себя строку, состоящую из строчных латинских букв. Корректный пароль к нему представляет собой логин с приписанными к нему справа какими-либо маленькими латинскими буквами (возможно, количество приписанных букв нулевое). То есть логин является префиксом подходящего ему пароля.

Ваша задача — отыскать соответствующие друг другу логины и пароли. Каждая строка из списка может быть использована в качестве логина или пароля ровно один раз.

Формат входных данных

В первой строке входного файла находится натуральное число n — количество аккаунтов ($1 \leq n \leq 10^5$).

В следующих $2 \times n$ строках дано по одной строке, состоящей из строчных латинских букв.

Суммарная длина строк не превышает 5×10^5 .

Формат выходных данных

В выходной файл выведите n строк, в каждой из которых два числа — первое из которых является индексом строки логина, а второе — индексом строки пароля.

Гарантируется, что ответ существует. Если ответов несколько, выведите любой.

Пример

стандартный ввод	стандартный вывод
2	3 4
abac	1 2
abacab	
aba	
abaa	

Задача B. AliKingspress

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	64 мегабайта

Для экстренного пополнения боеприпасов и вооружения агенты «Кингсман» пользуются службой «AliKingspress». Помимо запроса помощи с вооружением, можно также делать другие запросы, однако уже не бесплатно, а за бонусные баллы. Баллы можно получать каждый день, заходя в специальное приложение. В первый день пользователь получает a_1 баллов, во второй — a_2 баллов, ..., в n -й день — a_n баллов. После этого, заходя каждый день, пользователь будет все еще получать a_n бонусов. Если же пропустить один или несколько дней и не заходить в приложение, при следующем заходе начисление бонусов опять начнется с a_1 .

Эггси посчитал, что для выполнения всех дополнительных запросов, которые он хочет, нужно x бонусов. Так как он перфекционист, лишние бонусы ему не нужны, он хочет накопить их ровно x , ни больше, ни меньше. Однако сделать это нужно как можно быстрее, потому что долго ждать он не намерен. Задачу нахождения минимального количества дней, требуемого для этого, он поручил вам — своего верному программисту, пока он сам спасает мир. Помогите ему!

Формат входных данных

В первой строке содержится два числа n и x — количество различных бонусов, а также суммарное количество бонусов, которое нужно набрать Эггси ($1 \leq n \leq 100, 1 \leq x \leq 10^6$).

Во второй строке содержится n чисел a_i — размеры бонусов в зависимости от количества дней захода в приложение ($1 \leq a_i \leq 1\,000$).

Формат выходных данных

В единственной строке выведите минимальное количество дней, нужное для получения ровно x бонусов или -1 , если набрать ровно x бонусов невозможно.

Примеры

стандартный ввод	стандартный вывод
4 14 1 2 3 4	5
5 12 1 4 2 6 3	6
3 8 3 4 2	-1

Замечание

В первом примере Эггси может заходить в приложение 5 дней подряд и получить, соответственно, $1 + 2 + 3 + 4 + 4 = 14$ бонусов.

Во втором тестовом примере Эггси может зайти в приложение три дня подряд, затем пропустить один день, а затем зайти еще два дня подряд. В результате он получит $1 + 4 + 2 + 1 + 4 = 12$ бонусов и потратит на это $3 + 1 + 2 = 6$ дней.

Задача С. Протокол «Судного дня»

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

Согласно протоколу «Судного дня», выжившим агентам Кингсман нужно добраться до некоего винного магазина. Для этого они могут воспользоваться лондонским метро и секретными тоннелями Кингсман. Сеть лондонского метро представляет собой дерево, где вершины — станции, а ребра — перегоны между ними. Станции пронумерованы числами от 1 до n , и винный магазин находится на станции с номером 1. Все перегоны имеют одинаковую длину. Так как за выжившими агентами может вестись наблюдение, и перемещение по метро может раскрыть их местонахождение, они должны перемещаться по метро только в направлении от магазина, чтобы запутать противника. Другими словами, они могут проехать на метро от станции до соседней, только если расстояние по ребрам от магазина до первой станции меньше, чем до второй. Все тоннели кингсман соединяют две станции метро, такие, что от одной можно доехать до другой перемещаясь по метро только в направлении от магазина.

Мерлин прорабатывает различные сценарии, которые могут возникнуть при активации протокола «Судного дня». Каждый сценарий характеризуется тем, на какой станции метро находятся выжившие агенты, и каким наибольшим количеством тоннелей они могут воспользоваться. По метро агенты могут перемещаться сколько угодно, но только в направлении, удаляющем их от магазина. Мерлину необходимо для каждого сценария узнать, на каком минимальном расстоянии от магазина могут оказаться выжившие агенты, если будут следовать всем указаниям, ведь оставшийся путь им придется проделать пешком.

Формат входных данных

В первой строке дано одно целое число n — количество станций метро ($2 \leq n \leq 200\,000$). В следующих $n - 1$ строках дано по два целых числа, a_i и b_i — номера станций, соединенных перегонем метро ($1 \leq a_i, b_i \leq n$). В следующей строке дано одно целое число m — суммарное количество тоннелей кингсман ($0 \leq m \leq 200\,000$). В следующих m строках дано по два целых числа, u_i и v_i — номера станций, соединенных тоннелем кингсман ($1 \leq u_i, v_i \leq n, u_i \neq v_i$). Гарантируется, что от v_i можно добраться до u_i , перемещаясь только в направлении удаления от магазина. В следующей строке дано одно целое число q — количество сценариев ($0 \leq q \leq 200\,000$). В следующих q строках дано по два целых числа, s_i и k_i — номер станции метро, на которой изначально находятся выжившие агенты, и количество тоннелей кингсман, которыми они могут воспользоваться ($1 \leq s_i \leq n, 0 \leq k_i \leq m$).

Формат выходных данных

Для каждого сценария выведите одно число — номер станции, находящейся на минимальном расстоянии до магазина, среди тех, до которых могут добраться выжившие агенты.

Пример

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

Задача D. Плохая многозадачность

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

Спасать мир — задача не из легких, трудности возникают на каждом шагу. Так и сейчас, таки пробравшись в кабинет к Поппи Адамс, Эггси вновь попал в передрыгу — на экране появилась сама Поппи и сказала, что агент слишком предсказуем, и она как всегда на несколько шагов впереди, ведь запуск уничтожения антидота к вирусу, который она распространила, уже запущен, и ничто не может это остановить, мир будет уничтожен.

Однако Эггси не просто так является агентом лучшей секретной службы, поэтому он сразу же достал свой ноутбук, вычислил IP-адрес сервера, на котором запущена программа уничтожения и взломал его. Оказалось, что это личный компьютер Поппи, а она не очень хорошо в них разбирается. Ее операционная система MS ADOS не поддерживает параллельное исполнение программ, поэтому все программы исполняются в одном потоке. Для этого поддерживается очередь задач на исполнение, и раз в секунду первая задача из очереди запускается на исполнение. За секунду программа может успеть сделать не более b операций. Если программа успевает выполнить все необходимые операции, она удаляется из очереди, иначе она переносится в конец очереди на дальнейшее выполнение.

Эггси быстро был обнаружен, а на сервере сменили пароль, тем самым закрыв для него доступ к серверу. Однако перед этим он успел узнать, что программе для уничтожения нужно a_1 операций для завершения, а также добавить в очередь задач на исполнение $n - 1$ других задач с a_2, a_3, \dots, a_n операциями для выполнения соответственно. Теперь он хочет понять, сколько у него времени, чтобы добраться до Поппи, пока программа уничтожения не завершилась. Помогите ему найти количество секунд, которое потребуется программе уничтожения для завершения.

Формат входных данных

В первой строке содержится два числа n и b — суммарное количество программ, запущенных на сервере, и максимальное количество операций, которое может выполняться на сервере за секунду ($1 \leq n \leq 10^5, 1 \leq b \leq 10^9$).

В следующей строке содержится n чисел a_i — количество операций для выполнения программы уничтожения и добавленных Эггси программ соответственно в порядке, в котором они расположены в очереди ($1 \leq a_i \leq 10^9$).

Формат выходных данных

В единственной строке выведите количество секунд, через которое программа уничтожения завершится.

Примеры

стандартный ввод	стандартный вывод
5 1 3 4 3 1 2	10
4 2 6 2 5 1	7

Замечание

Во втором тестовом примере посекундная последовательность действий будет следующая:

- 1 секунда: первая программа (программа уничтожения) запускается на исполнение, выполняет 2 операции, а затем перемещается в конец очереди;
- 2 секунда: вторая программа запускается на исполнение, выполняет 2 операции, завершается и удаляется из очереди;

- 3 секунда: третья программа запускается на исполнение, выполняет 2 операции, а затем перемещается в конец очереди;
- 4 секунда: четвертая программа запускается на исполнение, выполняет 1 операцию, завершается и удаляется из очереди;
- 5 секунда: первая программа (программа уничтожения) запускается на исполнение, выполняет 2 операции, а затем перемещается в конец очереди;
- 6 секунда: третья программа запускается на исполнение, выполняет 2 операции, а затем перемещается в конец очереди;
- 7 секунда: первая программа (программа уничтожения) запускается на исполнение, выполняет 2 операции, завершается и удаляется из очереди.

Дальнейшее выполнение программ нас не интересует, как видно, через 7 секунд программа уничтожения завершится.

Задача Е. Разбиение на пары

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

Секретные агентства «Кингсман» и «Стейтсман» готовят совместную масштабную операцию. Для операции им необходимо разбить агентов на пары.

У каждого агента есть предпочтение. Он либо хочет быть в паре с коллегой из своего агентства, либо с агентом из другого. При этом, если агент получит в пару агента, не подходящего под свои предпочтения, то ему будет дискомфортно, и он не сможет работать с максимальной эффективностью.

Мерлин знает предпочтения всех агентов. Он хочет разбить их по парам так, чтобы минимизировать количество агентов, которым будет дискомфортно.

Формат входных данных

Входные данные содержат четыре натуральных числа a , b , c и d — предпочтения агентов: число агентов «Кингсман», которые хотят работать с коллегами, число агентов «Кингсман», которые хотят в напарники агента из «Стейтсман», число агентов «Стейтсман», которые хотят работать с коллегой из своего агентства и число агентов «Стейтсман», которые хотят быть в паре с агентом из «Кингсман», соответственно. ($1 \leq a, b, c, d \leq 100$).

Гарантируется, что $a + b + c + d$ делится на 2 без остатка.

Формат выходных данных

Выведите одно число — минимальное количество агентов, которым будет дискомфортно.

Примеры

стандартный ввод	стандартный вывод
1 1 1 1	2
2 1 2 1	0

Задача F. Гарри и носки

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

Не надо грязные носки
Забрасывать под шкаф!
Они черствеют от тоски,
В такую глушь попав.

Юнна Мориц, Стирать свои носки!

Даже секретным агентам порой приходится заниматься самыми обыкновенными вещами.

Однажды, ранним воскресным утром, Гарри Харт решил навести порядок в старинном особняке агентства Кингсман. Спустившись в подвал, он начал подметать пол и случайно обнаружил под шкафом большой чемодан, принадлежавший одному из его предшественников. Как известно, изначально люди из Кингсман были портными, поэтому в чемодане оказались n аккуратно разложенных пар носков. У каждого носка был свой цвет, и у носков в паре был один и тот же цвет и номер пары, которому он принадлежит. Пары пронумерованы различными целыми числами от 1 до n . Кроме того, на каждом носке была маркировка «Л» или «П», обозначающая для левой или правой ноги предназначен этот носок. В каждой паре ровно один носок для левой ноги и ровно один для правой.

Предприимчивый Гарри быстро понял, что носки с такой богатой историей будут отличным подарком для новобранцев, поэтому он решил постирать эти носки и отнести на рынок. Для того, чтобы носки наверняка отстирались, он сложил все носки в одну большую кучу и положил в стиральную машину. По окончании стирки носки выглядели изумительно, но вот беда — все носки оказались перемешаны. Гарри — человек ленивый, поэтому он решил как-то разбить носки на пары так, чтобы в каждой паре был один левый и один правый носок. Конечно, в результате этого носки в одной паре могли оказаться разных цветов. Гарри знает, что среди молодёжи сейчас популярно носить разных цветов на левой и на правой ногах, поэтому Гарри решил презентовать подарок, как «пара разноцветных носков». Однако, в результате перемешивания могла образоваться и пара с двумя носками одинаковых цветов, а в таком случае Гарри обидит кого-нибудь, чего он никак не может допустить.

Гарри решил оценить вероятность того, что он никого не обидит. Для этого ему необходимо знать, сколько есть способов разбить носки на пары так, чтобы в каждой паре был один носок для левой ноги и один носок для правой ноги, а также не было пар, в которых оба носка одного цвета. Два разбиения на пары считаются различными, если различаются множества пар, содержащиеся в них. Две пары носков называются различными, если номера пар, из которых взяты левые носки в этих парах, различаются или номера пар, из которых взяты правые носки, различаются. Обратитесь к примерам для лучшего понимания условия. Так как число таких способов может быть очень большим, Гарри интересуется остаток от деления количества способов на $10^9 + 7$.

Помогите Гарри, и он не останется в долгу.

Формат входных данных

В первой строке задано целое число n — количество пар носков ($1 \leq n \leq 3000$).

Во второй строке задано n целых чисел a_1, a_2, \dots, a_n , где a_i обозначает цвет носков в паре i ($1 \leq a_i \leq n$). Одинаковые цвета обозначены одинаковыми числами, разные — разными.

Формат выходных данных

Выведите одно целое число — остаток от деления числа способов на $10^9 + 7$.

Примеры

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

Замечание

В первом тестовом примере два способа разбить носки на пары выглядят так (первое число в скобках — номер пары, из которой взят левый носок, второе — правый):

- $\{(1, 2), (2, 3), (3, 1)\}$;
- $\{(1, 3), (2, 1), (3, 2)\}$.

Во втором тестовом примере четыре способа разбить носки на пары:

- $\{(1, 3), (2, 4), (3, 1), (4, 2)\}$;
- $\{(1, 4), (2, 3), (3, 1), (4, 2)\}$;
- $\{(1, 3), (2, 4), (3, 2), (4, 1)\}$;
- $\{(1, 4), (2, 3), (3, 2), (4, 1)\}$.

В третьем тестовом примере при любом разбиении будет пара, в которой оба носка будут иметь цвет 2, поэтому ответ — 0.

В четвёртом тестовом примере четыре разбиения выглядят так:

- $\{(1, 2), (2, 1), (3, 4), (4, 3)\}$;
- $\{(1, 3), (2, 1), (3, 4), (4, 2)\}$;
- $\{(1, 2), (2, 4), (3, 1), (4, 3)\}$;
- $\{(1, 3), (2, 4), (3, 1), (4, 2)\}$.

Задача G. Плеер Кингсманов

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

Каждому, кто принят в «Кингсман», выдают специальный плеер, чтобы они могли слушать музыку, пока находятся в пути на задание.

Гарри показал себя как лучший агент во множестве заданий. Он умеет пользоваться огромным числом технологий, но, к сожалению, за все время работы в агенстве он так и не узнал, на какую кнопку нужно нажимать, чтобы песни можно было проигрывать в случайном порядке.

Перед тем, как отправиться на задание в очередной раз, он увидел, как другие агенты нажимают на одну из кнопок в центре плеера. Гарри тоже нажал на одну из них. Известно, что нажатие на первую кнопку запускает последовательное воспроизведение песен с той, на которой плеер был остановлен (если текущая песня — последняя в списке треков, следующей будет проигрываться первая песня из списка), а вторая кнопка запускает воспроизведение песен в случайном порядке.

Гарри слишком занят мыслями о предстоящем деле, поэтому не может проверить, проигрываются ли песни в случайном порядке или нет. Помогите Гарри — определите номер позиции первой песни, послушав которую, Гарри поймет, что треки проигрываются случайном порядке, либо определите, что песни проигрываются в последовательном порядке.

Формат входных данных

В первой строке входных данных содержится единственное целое число n — количество песен в плеере ($3 \leq n \leq 10^5$).

Во второй строке содержатся n различных целых чисел a_i ($1 \leq a_i \leq n$), где i -е число обозначает номер песни, которая играла i -й.

Формат выходных данных

Если песни проигрываются в последовательном порядке, в единственной строке выведите «NO» без кавычек. В противном случае на первой строке выведите «YES», а на следующей за ней номер позиции первой песни, послушав которую, можно понять, что Гарри слушает песни в случайном порядке.

Примеры

стандартный ввод	стандартный вывод
3 1 2 3	NO
3 1 3 2	YES 2

Задача Н. Башни

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

Ричмонд Валентайн построил башни сотовой связи, через которые он планирует передавать сигнал на SIM-карты его компании. Теперь он хочет найти территорию, которую покрывают его башни.

Карта Земли представляет собой клетчатую таблицу n на m . Башни занимают ровно одну клетку в этой таблице. Назовём фигуру из клеток *связной*, если из каждой клетки этой фигуры можно дойти до всех остальных, идя только через границу клеток и проходя только по клеткам этой фигуры. Назовём связную фигуру *клеточно-выпуклой*, если для каждой вертикальной и горизонтальной прямой, фигура пересекает эту прямую в не более чем одном непрерывном отрезке. Тогда Ричмонд Валентайн считает, что башни покрывают минимальную связную клеточно-выпуклую фигуру, которая содержит все построенные им башни. Теперь он хочет найти область, которую покрывают его башни.

Формат входных данных

В первой строке входного файла находятся целые числа n и m — размеры Земли ($1 \leq n, m \leq 500$). В следующих n строчек по m символов находится описание карты Земли. Если в клетки (i, j) находится башня, то j -й символ i -й строки равен «*», иначе этот символ равен «.».

Формат выходных данных

Выведите n строк по m символов в каждой. Если клетка (i, j) лежит в покрытой области, то j -й символ i -й строки должен быть равен «*», иначе он должен быть равен «.».

Если ответов несколько, выведите любой.

Пример

стандартный ввод	стандартный вывод
3 3	***
***	***
.	***
.	