

Задача А. Акромантулы

Имя входного файла:	acromantula.in
Имя выходного файла:	acromantula.out
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

Акромантул (Acromantula) — это огромный восьмиглазый паук, который умеет говорить. Акромантул — хищник и предпочитает крупную добычу. Он плетёт свою паутину в форме купола на поверхности земли. Женская особь крупнее, чем мужская, и может откладывать до сотни яиц за один раз. Яйца акромантула белые и мягкие, размером с надувной детский мяч. Детёныши вылупляются через 6-8 недель. Яйца акромантула помещены Отделом по контролю волшебных существ в Класс А: «Товары не подлежащие продаже». Это означает, что за импорт яиц акромантула или торговлю ими полагается суровое наказание.

— «Фантастические звери и места их обитания»

Магический зоолог Ньют Саламандер изучает колонию акромантулов. Сейчас его интересует их размножение. Он уже знает, что самки акромантулов откладывают яйца, из которых вылупляются детёныши. Ньют хочет опубликовать родословную этой колонии в журнале о волшебных существах: для каждого паука он хочет выяснить, кто является его матерью. Конечно, ему хотелось бы ещё узнать отцов, но процессы оплодотворения у акромантулов на данный момент плохо изучены, поскольку все, кто пытался проследить за этим процессом, были немедленно съедены; поэтому ничего об отцовстве в этой колонии не известно.

Ему известно, что вся колония началась с одной самки, а все остальные особи являются её потомками, и с самого начала никто не покидал колонию и не умирал. Про каждого из пауков Ньют знает возраст. Разумеется, возраст детёныша всегда строго меньше возраста матери. Кроме того, Ньют внимательно изучил каждую особь и оценил, какое наибольшее количество яиц она могла отложить.

Помогите зоологу построить такую родословную, чтобы она не противоречила известной ему информации:

- Возраст детёныша меньше возраста матери.
- Про каждого акромантула известно наибольшее возможное количество детёнышей.

Формат входных данных

В первой строке входного файла задано одно целое число n — количество акромантулов в колонии ($1 \leq n \leq 2 \cdot 10^5$).

В следующих n строках задано по два числа: a_i и c_i , которые означают, что возраст i -й особи равен a_i , и у неё может быть не более c_i детёнышей ($1 \leq a_i \leq 10^9$, $0 \leq c_i \leq n - 1$).

Формат выходных данных

Если родословной построить невозможно, выведите «NO» (без кавычек).

В противном случае выведите в первой строке «YES» (без кавычек), а во второй — n чисел: i -е из них равно номеру матери i -го паука, или 0, если это первая особь в колонии. Особи нумеруются с единицы в том порядке, в котором они заданы во входном файле.

Если существует несколько возможных ответов, выведите любой из них.

Примеры

acromantula.in	acromantula.out
5 5 4 4 1 1 0 2 0 3 0	YES 0 1 2 1 1
3 2 1 2 0 5 1	NO

Задача В. Волшебный чемодан

Имя входного файла: `array.in`
Имя выходного файла: `array.out`
Ограничение по времени: 1.5 секунд
Ограничение по памяти: 64 мегабайта

В чемодане Ньюта Саламандера находится n волшебных животных и столько же отделений для них. Отделения пронумерованы от 1 до n . В каждое отделение помещается ровно одно животное. У каждого существа есть уровень опасности t . Как известно, животные весьма активны. Они не любят находиться в одном месте и поэтому постоянно меняются местами внутри чемодана.

Назовём группу подряд идущих животных, которые находятся в отделениях с l по r , отрезком. В течение m минут происходило два типа событий:

1. Два непересекающихся равных по размеру отрезка животных меняются местами. Отрезок с l_1 по r_1 меняется с отрезком с l_2 по r_2 . Формально, животное в ячейке i меняется местом с животным в ячейке $i - l_1 + l_2$, где i от l_1 до r_1 .
2. Зоолог спрашивает количество животных на отрезке с l по r , у которых сила t не меньше, чем a и не больше, чем b .

Вы узнали, какие события происходили в чемодане, а также начальное положение животных. Найдите ответы на вопросы учёного.

Формат входных данных

В первой строке задано число n и m — количество клеток и количество запросов ($1 \leq n \leq 10^6$; $1 \leq m \leq 2 \cdot 10^3$).

Во второй строке задана последовательность чисел t_i длины n — силы животных ($1 \leq t_i \leq 10^9$). Далее следует m строк. В каждой записаны числа $type$ — тип запроса ($1 \leq type \leq 2$).

Если $type = 1$, то далее следуют числа l_1, r_1, l_2, r_2 — запрос обмена местами животных. Гарантируется, что отрезки не пересекаются и имеют равную длину ($1 \leq l_1 \leq r_1 \leq n$, $1 \leq l_2 \leq r_2 \leq n$).

Если $type = 2$, то далее следуют числа l, r, a, b — запрос зоолога ($1 \leq l \leq r \leq n$, $1 \leq a \leq b \leq 10^9$).

Формат выходных данных

Для каждого запроса второго типа вывести количество подходящих животных.

Примеры

array.in	array.out
3 2 1 2 3 1 1 1 3 3 2 2 3 1 2	2
6 5 1 2 3 4 5 6 1 1 3 4 6 1 2 3 5 6 2 3 6 2 5 1 1 1 5 5 2 2 5 2 6	2 3

Задача С. Волшебные существа

Имя входного файла: `beasts.in`
Имя выходного файла: `beasts.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Как известно, у одного из самых известных магозоологов Великобритании Ньюта Саламандера имеется волшебный чемодан. На чемодан Ньют наложил Заклинание незримого расширения, а внутри него хранил обширную коллекцию редких, находящихся под угрозой исчезновения волшебных существ, найденных им во время кругосветных путешествий.

Однажды Ньют оставил свой чемодан полуоткрытым, и существа решили совершить побег. Однако сделать это одновременно они не могут, так как створки чемодана очень узкие. Время, необходимое каждому питомцу для того, чтобы вылезти из чемодана, составляет s . Первый из питомцев покидает чемодан в момент времени t , а последующие — в моменты времени $t + s$, $t + 2 \cdot s$ и так далее.

Ньют Саламандер слишком поздно узнал о хитром плане питомцев, и сейчас его интересует, какое суммарное количество существ сбежало из чемодана в промежутки времени $[a_i, b_i]$, включая границы. Питомцев в волшебном чемодане содержится бесконечное количество.

Формат входных данных

В первой строке входного файла заданы числа t и s — момент времени, когда первый питомец покинул чемодан, и интервал времени между побегами питомцев соответственно ($0 \leq t \leq 10^{12}$, $1 \leq s \leq 10^{12}$).

Во второй строке содержится число n — количество интересующих Саламандера отрезков времени ($1 \leq n \leq 100\,000$). В следующих n строках содержатся числа a_i, b_i — левая и правая границы i -го отрезка времени ($0 \leq a_i, b_i \leq 10^{12}$).

Формат выходных данных

В выходном файле выведите единственное число — суммарное количество существ, сбежавших во время данных промежутков времени. Ответ для каждого промежутка считается независимо от других промежутков.

Примеры

beasts.in	beasts.out
4 3 2 1 3 5 13	3
1 10 3 1 1 1 1 1 1	3

Замечание

В первом тесте из условия существа сбегает в моменты времени 7, 10 и 13, принадлежащие второму промежутку времени. Во время первого промежутка ни одно существо не совершает побег.

Задача D. Кольцевые дороги

Имя входного файла: `circles.in`
Имя выходного файла: `circles.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Дорожная сеть Нью-Йорка состоит из двух кольцевых дорог, являющихся концентрическими окружностями, и нескольких дорог, соединяющих их. Введем систему координат так, что центр окружностей совпадает с началом координат, а ось ОУ направлена на север. Первая кольцевая дорога имеет радиус 10 километров, а вторая — 20. По кольцевым дорогам можно двигаться в любую сторону. Дороги, соединяющие их, являются односторонними, и по ним можно проехать только от внутреннего кольца до внешнего. Всего дорог, соединяющих кольца, n штук, они задаются углами ang_i , i -й дорогой является заключенный между окружностями отрезок луча, имеющего угол наклона ang_i от оси ОХ.

Сейчас Ньют продумывает план поимки Ньюля. Помогите ему, ответьте на q его вопросов. Вопрос номер j задается двумя числами: a_j и b_j , он значит, что Ньют хочет узнать кратчайшее расстояние от точки на первом кольце, такой, что угол наклона отрезка, проведенного в нее из начала координат, равен a_j , до точки на втором кольце, имеющей аналогичный угол наклона равный b_j .

Формат входных данных

В первой строке даны два числа n и q — количество дорог, соединяющих кольцевые дороги, и количество вопросов Ньюта, соответственно ($1 \leq n, q \leq 10^5$).

В следующих n строках дано по одному вещественному числу ang_i — угол наклона луча, соответствующего i -й дороге, в градусах ($0 \leq ang_i < 360$).

В следующих q строках дано по два вещественных числа a_j и b_j — j -й вопрос Ньюта ($0 \leq a_j, b_j < 360$).

Все углы даны в градусах. Все вещественные числа даны с не более чем 6 цифрами после запятой.

Формат выходных данных

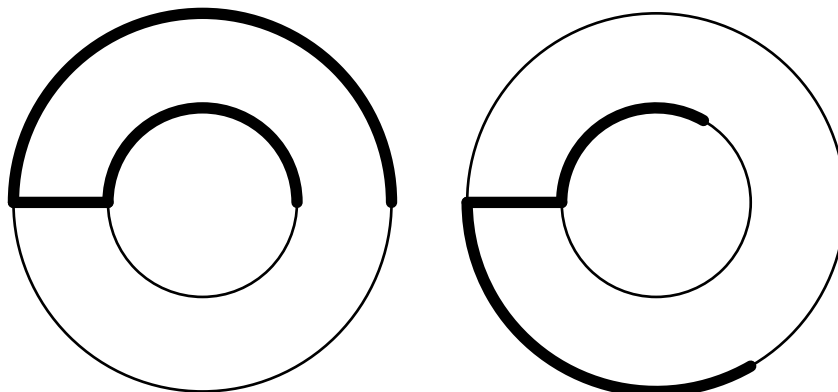
На каждый вопрос выведите в новой строке одно вещественное число — кратчайшее расстояние между данными точками, абсолютная или относительная погрешность не должна превышать 10^{-6} .

Пример

<code>circles.in</code>	<code>circles.out</code>
1 2	104.2477796077
180	72.8318530718
0 0	
60 300	

Замечание

Пояснение к тесту из примера. Кратчайший путь для первого и второго вопроса соответственно.



Задача Е. За коллективизм!

Имя входного файла: `collectivism.in`
Имя выходного файла: `collectivism.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Магические твари сбежали! Ньют Саламандер вместе со своими n помощниками искали их повсюду и наконец-то нашли.

Теперь магический закон гласит, что все помощники должны представить отчет в МАКУСА (в «Магический Конгресс Управления по Северной Америке»), в нем следует указать, сколько тварей поймал каждый из них. После этого каждый помощник получит награду (подразумевается, что награда будет тем выше, чем больше тварей поймал помощник). Дабы поддержать коллективный дух, Ньют хочет, чтобы награды помощников были равны. Поэтому он решил, что некоторые из помощников заберут себе несколько тварей так, чтобы в отчете у всех значилось одинаковое количество тварей.

Было решено, что суммарное количество "изъятых" тварей не должно превышать некоторое число k , иначе конгресс может заподозрить неладное. Чтобы данное условие выполнялось, Саламандер решил, что некоторые из его помощников не будут участвовать в отчетности. Разумеется, он хочет минимизировать их количество. Помогите ему в этом.

Формат входных данных

В первой строке входного файла дано два натуральных числа n и k ($1 \leq n \leq 10^5$, $1 \leq k \leq 10^9$) — количество помощников и максимальное количество тварей, которое разрешено взять. Во второй строке содержатся n натуральных чисел, где i -ое число a_i ($1 \leq a_i \leq 10^9$) означает, что i -ый помощник поймал a_i тварей.

Формат выходных данных

В первой строке выведите число m — количество людей, которые не будут представлены к отчету. Во второй строке выведите m чисел — номера этих людей в возрастающем порядке.

Пример

collectivism.in	collectivism.out
6 8	3
8 15 38 2 1 25	2 3 6

Задача F. Кроссворды

Имя входного файла: `crosswords.in`
Имя выходного файла: `crosswords.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Саламандер очень любит своих существ и поэтому решил скрасить им времяпровождение в чемодане. Он решил каждому существу дать по кроссворду, однако их много, и кроссвордов соответственно нужно тоже достаточное число. Саламандер решил, что для простоты каждый кроссворд будет состоять ровно из 4 слов — 2 по вертикали и 2 по горизонтали, а также у них будет ровно 4 попарных пересечения, образующих прямоугольник, возможно нулевой площади.

Саламандер уже придумал 4 различных слова для нового кроссворда, теперь он хочет знать количество различных кроссвордов, которые из них можно составить. Он считает, что два кроссворда различны, если их нельзя наложить друг на друга так, чтобы они полностью совпали. Помогите ему посчитать нужное ему количество способов.

Формат входных данных

В i -й из 4 строк содержится w_i — i -е слово, придуманное Саламандром ($2 \leq |w_i| \leq 30$). Гарантируется, что каждое слово состоит только из строчных букв латинского алфавита и что все слова попарно различны.

Формат выходных данных

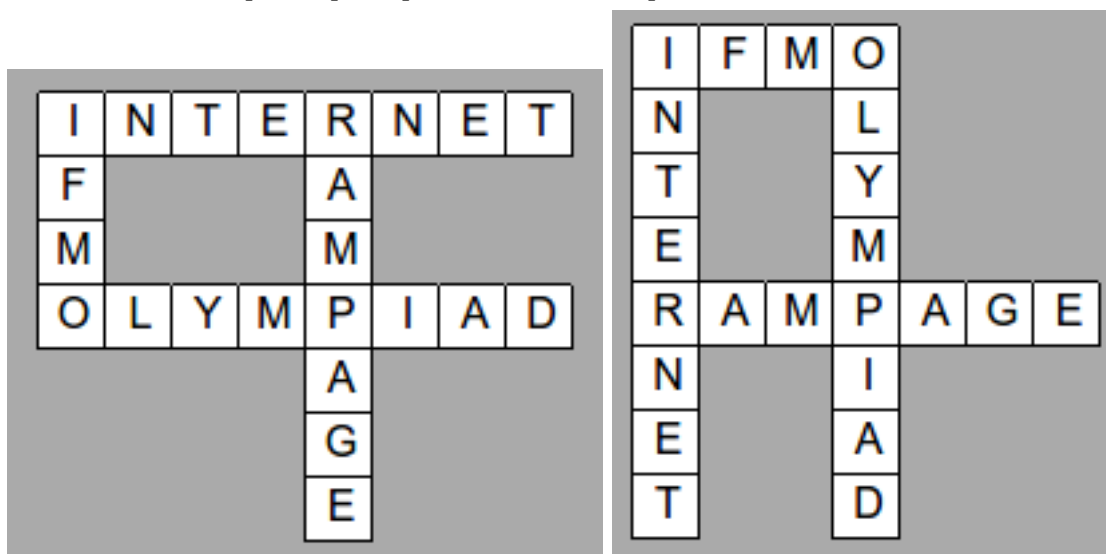
В единственной строке выведите количество способов составить кроссворд из данных слов. Учтите, что слова по горизонтали записываются слева направо, а слова по вертикали — сверху вниз.

Пример

crosswords.in	crosswords.out
internet ifmo rampage olympiad	2

Замечание

Оба возможных кроссворда представлены на картинках:



Задача G. Взрывопотам

Имя входного файла: `deepcycle.in`
Имя выходного файла: `deepcycle.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Стоило Ньюту немного отвлечься от присмотра за своим зверинцем, как у него сразу сбежал взрывопотам. Ньют должен поймать его как можно быстрее, пока он не разнес половину города.

Взрывопотамы — странные существа, их поведение может озадачить или даже напугать человека, не видевшего их раньше. На авеню, на которой Ньют будет ловить взрывопотама, стоят в ряд n фонарных столбов разной высоты. Волшебник собирается встать около одного из них и приманить взрывопотама. Разъяренный взрывопотам прибежит к этому столбу и начнет крушить все большие столбы, который он увидит. Точнее, он сломает столб около Ньюта, побежит к ближайшему справа столбу, который строго выше чем столб Ньюта, сломает его и продолжит так же ломать столбы справа, ломая ближайший справа столб, который выше последнего сломанного.

Ньют хоть и рассеянный, но магией владеет хорошо. Особенно хорошо ему удаются пространственные преобразования: он может применить заклинание, которое перенесет несколько самых левых столбов в конец улицы, поставив их в таком же порядке после последнего столба. Например, если на улице стояли столбы высотой 2, 4, 1, 3, 3, и чародей применит заклинание к первым двум столбам, то после этого столбы будут стоять на улице в порядке 1, 3, 3, 2, 4.

Чем больше столбов снесет взрывопотам, тем сильнее он устанет, и его будет проще поймать. Помогите Ньюту определить, какое максимальное количество столбов сломает взрывопотам, если волшебник может один раз перенести несколько столбов из начала улицы в конец и после этого встать около любого столба.

Формат входных данных

В первой строке дано одно целое число n — количество фонарей на авеню ($1 \leq n \leq 200\,000$).

В следующей строке дано n целых чисел a_i — высоты фонарей в порядке от начала авеню к концу ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Выведите одно целое число — максимальное число столбов, которое может сломать взрывопотам.

Примеры

<code>deepcycle.in</code>	<code>deepcycle.out</code>
5 2 4 1 3 3	3
6 2 5 3 5 1 5	2

Задача Н. Восстановление числа

Имя входного файла: `numrecovery.in`
Имя выходного файла: `numrecovery.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Мистер Саламандер едет в поезде. Он записал два числа n и m и отлучился. Когда он вернулся, оказалось, что его чай залил некоторые цифры числа n .

Мистер Саламандер любит гоголомки, поэтому ему стало интересно: какой наименьший остаток от деления на m могло давать исходное число n ?

Вам дана строка, представляющее залитое число n , в которой на каждой позиции находится либо цифра, либо знак вопроса, обозначающий, что данная цифра залита чаем, и модуль m .

Формат входных данных

В первой строке находится непустая строка n и натуральное число m — модуль ($1 \leq m \leq 10^9$). Строка n состоит из цифр и знаков вопроса, не имеет ведущих нулей, и ее длина не превосходит 14.

Формат выходных данных

В единственной строке выведите наименьший остаток от деления, который можно получить, заменив знаки вопросов на цифры, так, чтобы получившееся число не имело ведущих нулей.

Примеры

<code>numrecovery.in</code>	<code>numrecovery.out</code>
3?1? 3215	0
?? 20	0
?1? 730	80

Задача I. Сверкающие плюсы

Имя входного файла: `pluses.in`
Имя выходного файла: `pluses.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 512 мегабайт

Проказливый нюхль — маленький, пушистый черный зверек с вытянутой закругленной мордочкой, похожий на помесь крота и утконоса. Нюхль падок на все блестящее: он украдет или стянет любой сверкающий предмет, который попадется ему на глаза.

На этот раз зверек нашел матрицу $n \times m$, состоящую из ярко сверкающих единичек и неинтересных ему нулей. Нюхль очень хочет украсть как можно больше единиц, но матрица устроена таким образом, что стянуть он может только единицы, стоящие в матрицы в виде плюса. Plusом размера $4k + 1$ называется единица и отходящие от нее вправо, влево, вверх и вниз непрерывные последовательности единиц длины k (возможно, $k = 0$).

Помогите зверьку украсть как можно больше единиц! Найдите в данной матрице плюс наибольшего размера.

Формат входных данных

В первой строке входного файла заданы числа n и m — количество строк и столбцов матрицы соответственно ($1 \leq n, m \leq 5000$).

В каждой из последующих n строк содержится m символов, каждый из которых 0 или 1.

Формат выходных данных

В первой строке выходного файла выведите размер наибольшего плюса. В следующей строке выведите номер строки и номер столбца, в которых расположена центральная единица найденного плюса. Если ответов несколько, выведите ответ с наименьшим номером строки. Если ответов по-прежнему несколько, выведите ответ с наименьшим номером столбца.

Если в матрице отсутствуют плюсы, выведите -1.

Примеры

<code>pluses.in</code>	<code>pluses.out</code>
6 7 0000000 0001000 0001000 0111110 0001000 0001000	9 4 4
3 3 010 000 001	1 1 2

Задача J. Тайные комнаты

Имя входного файла: `rooms.in`
Имя выходного файла: `rooms.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

К Мистеру Саламандеру в руки попал план некоторого здания. На этом плане n тайных комнат, пронумерованных от 1 до n . Из каждой комнаты существует ровно один выход. Выход из i -й комнаты ведет в a_i -ю комнату.

Мистер Саламандер любит гулять по кругу, но с текущим планом это может ему не удастся. К тому же, он любит некоторый эффект неожиданности, поэтому хочет изменить выход ровно одной комнаты, на некоторый другой, отличный от изначального, чтобы существовал такой циклический маршрут, который начинается в комнате номер 1, посещает все вершины и не посещает никакую вершину дважды.

Помогите ему выяснить, можно ли это сделать.

Обратите внимание, что так как комнаты волшебные, выход из комнаты может вести в нее же саму, то есть $a_i = i$.

Формат входных данных

В первой строке находится натуральное число n — количество комнат ($2 \leq n \leq 10^5$).

В следующей строке находится n натуральных чисел a_i — в какую комнату ведет выход из комнаты с номером i ($1 \leq a_i \leq n$).

Формат выходных данных

В первой строке выведите два числа ($1 \leq x, y \leq n, x \neq y$) — номер комнаты, в которой нужно изменить выход, и номер комнаты, в который должен вести новый выход из комнаты с номером x . Новый выход не должен совпадать со старым, то есть должно выполняться условие $a_x \neq y$. Если таких ответов несколько — выведите любой.

Если сделать этого невозможно — выведите -1 -1.

Примеры

<code>rooms.in</code>	<code>rooms.out</code>
3 1 2 3	-1 -1
3 1 3 1	1 2
4 2 3 4 1	-1 -1

Задача К. Новый чемодан

Имя входного файла: `suitcase.in`
Имя выходного файла: `suitcase.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Мистер Саламандер решил, что ему нужен новый чемодан для волшебных существ. Как известно, чемодан тоже волшебный и производится не из обычных вещей.

У Саламандера есть n волшебных прутиков длиной $1, 2, \dots, n$ соответственно. Для создания чемодана нужно сложить из некоторых палочек прямоугольник с положительными сторонами, не ломая ни одну из палочек — иначе они перестанут быть волшебными. Волшебная сила чемодана оценивается как сумма длин палочек, из которых составлен прямоугольник. Саламандер хочет сделать как можно более мощный чемодан, и просит вас найти эту максимальную мощность. Помогите ему с этой задачей.

Формат входных данных

В первой и единственной строке содержится число n — количество волшебных палочек ($1 \leq n \leq 10^9$).

Формат выходных данных

В единственной строке выведите максимальную возможную силу чемодана. Если ни одного чемодана из этих палочек собрать не получится, в единственной строке выведите 0.

Примеры

<code>suitcase.in</code>	<code>suitcase.out</code>
7	28
1	0

Замечание

В первом тестовом примере можно собрать прямоугольник со сторонами $1 + 2, 3, 4 + 7$ и $5 + 6$ — суммарная длина палочек в этом прямоугольнике равна 28.

Во втором тестовом примере ни одного прямоугольника составить нельзя.