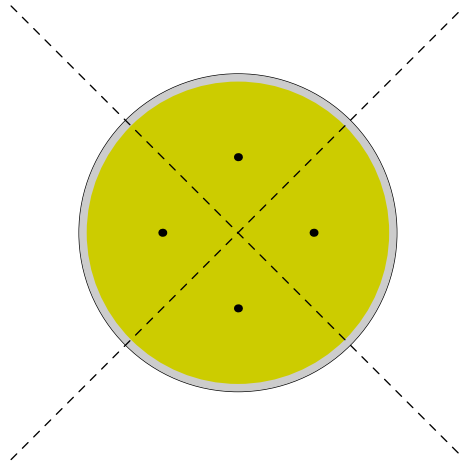


Задача А. Тортик

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

Винни-Пух, Кролик и Пятачок пришли в гости к Кристоферу Робину на его день рождения. Главный подарок — огромный круглый медовый торт с четырьмя свечками. Однако, чтобы съесть торт, его нужно сначала разрезать на четыре части. Зная, как сильно Винни-Пух любит мед, можно было ожидать, что он сразу же заберет себе самый большой кусок торта и наверняка застрянет в дверях, как это случилось, когда он и Пятачок ходили в гости к Кролику.



Чтобы избежать этого Кристофер Робин решил разрезать торт на четыре одинаковые части, а чтобы подчеркнуть торжественность, ведь это не обычный торт, а праздничный, и не усложнять разрезание, торт нужно разрезать двумя перпендикулярными разрезами, проходящими через его центр, причем на каждом из получившихся кусков должно быть ровно по одной свечке. При этом, чтобы никто не получил нецелую свечку и не обиделся, Кристофер Робин хочет, чтобы разрезы не проходили ни через одну свечку.

Формат входного файла

Во входном файле содержится описание торта. Центр торта принят за начало координат. Первая строка содержит одно целое число R ($1 \leq R \leq 10^9$) — радиус торта. В следующих четырех строках содержатся координаты свечек x_i, y_i . Гарантируется, что все свечки находятся строго в тортe.

Формат выходного файла

В выходной файл выведите «YES», если Кристофер Робин может разрезать торт так, как он хочет, и «NO» — если не может.

Примеры

<code>cake.in</code>	<code>cake.out</code>
10 0 1 1 0 0 -1 -1 0	YES

Задача В. Конфетки

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

У Кролика день рождения! Он пригласил в гости n гостей. Чтобы гостям не было грустно и скучно, Кролик купил n коробок конфет. Кролик любит разнообразие, поэтому конфеты были разные. В i -й коробке лежало a_i конфет.

В назначенный день с самого утра к Кролику начали приходить гости. Каждый гость характеризуется своей наглостью b_i . Это означает, что, зайдя домой к Кролику и увидев коробки конфет, он брал из каждой коробки, в которой не меньше, чем b_i , конфет, по одной и съедал её. Например, у Винни-Пуха вполне могла быть наглость один. Это значит, что он бы съел по конфете из каждой коробки.

Вечером, когда гости разошлись, Кролику стало интересно, кто съел сколько конфет. Помогите ему определить это.

Формат входного файла

В первой строке задано целое число n ($1 \leq n \leq 100\,000$) — количество коробок конфет. В следующей строке задано n натуральных чисел a_i ($1 \leq a_i \leq 10^9$) — сколько конфет в каждой коробке.

Далее, в следующей строке задано число m ($1 \leq m \leq 100\,000$) — количество гостей. В четвёртой и последней строке задано m чисел b_i ($1 \leq b_i \leq 10^9$) — наглости гостей.

Формат выходного файла

В выходной файл выведите n строк, i -ая из которых должна содержать количество конфет съеденных i -ым гостем.

Примеры

<code>candies.in</code>	<code>candies.out</code>
3	3
3 1 1	1
2	
1 1	

Задача С. Враг моего врага — мой друг!

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

Винни-Пух зарегистрировался в новой социальной сети, которая называется ВЛесу. В этой социальной сети у каждого пользователя, кроме списка его друзей, был также список его врагов. В этот список можно было добавить любого пользователя, но при этом удовлетворялись некоторые условия:

1. Если пользователь v является врагом пользователя u , то u **не обязательно** является врагом v
2. Пользователь не может быть врагом самого себя

Винни-Пуху очень понравилась эта социальная сеть. Он целыми днями сидел и записывал, кто же становился чьим врагом, так как хотел знать все, что происходит в их лесу. Он считал, что никто не пойдет в гости к своему врагу. Также, по его мнению, враг врага является другом, а любая уважающая своих друзей персона должна пойти в гости к своему другу. Винни-Пуху очень интересно узнать — сколько же у пользователя под номером v друзей. Пользователь u является другом пользователя v по версии Винни-Пуха, если выполняются некоторые условия:

1. u является врагом некоторого врага v
2. u не является врагом v

Заметим также, что никакой пользователь сам не является своим другом.

Формат входного файла

В первой строке входного файла задано числа n и m ($1 \leq n, m \leq 2000$) — количество пользователей, зарегистрированных в социальной сети и количество запросов соответственно.

В следующих m строках заданы запросы двух видов:

1. $+ v u$ — пользователь v начал считать пользователя u своим врагом
2. $? v$ — узнать количество друзей пользователя v по версии Винни-Пуха

Гарантируется, что входные данные корректны — пользователь не начнет считать себя своим врагом и никакой пользователь не станет врагом другого более одного раза.

Формат выходного файла

Для каждого запроса $? v$ выведите одно целое число — ответ на него в отдельной строке.

Примеры

<code>enemy.in</code>	<code>enemy.out</code>
5 5	1
+ 1 2	
+ 2 4	
+ 2 5	
+ 1 5	
? 1	

Задача D. Ямы

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

У Винни-Пуха большая радость — он купил автомобиль. И теперь, чтобы опробовать покупку, он собрался съездить на нем за медом.

От дома Винни-Пуха до дерева с правильными пчелами, дающими правильный мед, ведет дорога длиной n километров. Казалось бы, скорость передвижения на автомобиле там, где живет Винни, не регламентирована, и ехать можно сколь угодно быстро. Однако, если бы все было так просто, Винни бы не стал обращаться к Вам.

В некоторых местах на дороге расположены ямы. И, если Винни-Пух хочет сохранить свое новое авто в целости и сохранности (а он, конечно же, хочет), проезжать километр номер i , на котором расположена яма, нужно со скоростью, не превышающей a_i километров в час. Конструкция автомобиля такова, что после каждого проеханного километра он может мгновенно ускориться на один километр в час, замедлиться на эту же скорость или оставить скорость своего движения без изменений. При этом после изменения он будет ровно километр ехать с новой скоростью, после чего снова сможет ее изменить.

Еще одна особенность автомобиля заключается в следующем: он не может мгновенно разогнаться до нужной скорости, зато может мгновенно остановиться (выбросив тормозной парашют). Соответственно, первый километр пути он всегда проедет со скоростью 1 километр в час, а скорость на последнем километре может быть ограничена только наличием на этом километре ям.

Винни любит быструю езду, а еще больше он любит мед. Соответственно, добраться от своего дома до дерева он хочет как можно быстрее. Вас же он попросил определить, за сколько он сможет добраться до дерева и при этом сохранить целым автомобиль.

Формат входного файла

В первой строке входного файла дано одно целое число n ($1 \leq n \leq 10^5$) — длина дороги. В следующей строке дано n чисел a_i ($0 \leq a_i \leq 10^5$) — описание дороги. Если a_i равно 0, это означает, что соответствующий километр дороги ровный и его можно проезжать с любой скоростью. Иначе километр содержит яму и его можно проехать со скоростью, не превышающей a_i километров в час.

Формат выходного файла

Выведите в выходной файл одно вещественное число — минимальное количество часов, за которое Винни-Пух доедет до дерева. Ответ должен отличаться от правильного не больше, чем на 10^{-6} .

Примеры

holes.in	holes.out
7 0 0 2 0 0 0 1	4.16666666
8 0 0 0 0 0 0 2 0	3.5

Задача Е. Прыгать!

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

Тигра любит прыгать! К сожалению, прыгать по лесу не очень удобно — и за кочку можно зацепиться и в дерево врезаться. Умный и добрый Кролик решил помочь Тигре и построить несколько удобных дорог из желтого кирпича (ведь желтый — любимый цвет Тигры). Для этого он составил полную карту Стоакрового Леса и посчитал какие дороги можно построить, и сколько кирпича на каждую из них потребуется. Оценив запасы кирпича, Кролик понял что возможно вместо некоторых дорог можно построить настоящие лесные автобаны — широкие дороги, где Тигра сможет разогнаться еще быстрее и получить больше удовольствия от движения. Для постройки автобана вместо обычной дороги потребуется в c раз больше кирпича, независимо от расположения дороги.

Каждая дорога соединяет какие-то два интересных для Тигры места — домики обитателей Леса, поляны, где он может вдоволь попрыгать и так далее. Кролик пронумеровал интересные места целыми числами от 1 до n и выписал список возможных дорог. Теперь он хочет построить дорожную сеть таким образом, чтобы было построено как можно больше автобанов, но при этом можно было из каждого интересного места попасть в любое другое. Для реализации этого плана у Кролика в распоряжении есть k кирпичей. Помогите ему спланировать постройку дорог!

Формат входного файла

Первая строка входного файла содержит четыре целых числа n , m , k и c ($1 \leq n, m \leq 100000$; $1 \leq k \leq 10^9$; $1 \leq c \leq 1000$) — количество интересных мест, возможных дорог, общее количество кирпичей и коэффициент, во сколько раз больше кирпичей требуется на постройку автобана, соответственно.

Следующие m строк содержат описания дорог — три целых числа a_i , b_i , l_i ($1 \leq a_i, b_i \leq n$; $1 \leq l_i \leq 10^6$) — номера интересных мест, которые соединяет дорога и количество кирпича, необходимое на постройку этой дороги, соответственно.

Каждая дорога соединяет два различных интересных места. Между двумя интересными местами может быть более одной дороги.

Формат выходного файла

Если невозможно построить такую сеть дорог, то выведите в выходной файл единственное слово «Impossible».

Иначе, в первой строке выходного файла выведите два целых числа p и q — количество обычных дорог и автобанов в оптимальном плане. Во второй строке выведите p целых чисел — номера обычных дорог, которые необходимо построить в возрастающем порядке. В третьей строке выведите q целых чисел — номера автобанов, также в возрастающем порядке.

Дороги пронумерованы в порядке появления во входном файле. В случае нескольких оптимальных решений, выведите любое.

Примеры

<code>jump.in</code>	<code>jump.out</code>
4 2 10 2 1 2 3 3 4 5	Impossible
4 4 10 2 1 2 3 3 4 5 1 3 1 3 2 1	1 2 2 3 4

Задача F. Восстановление перестановки

Имя входного файла: `restore.in`
Имя выходного файла: `restore.out`
Ограничение по времени: 2 seconds
Ограничение по памяти: 256 megabytes

Сегодня в школе Кристофер изучал последовательности и перестановки. Напомним, что *перестановкой* чисел от 1 до n называется последовательность a_1, \dots, a_n , в которую каждое из указанных чисел входит ровно один раз.

Особенно ему понравились следующие определения:

- *Спуском* в позиции i в перестановке $\langle a_1, a_2, \dots, a_n \rangle$ называют такую ситуацию, что $a_i > a_{i+1}$;
- *Неподвижной точкой* в позиции i в перестановке $\langle a_1, a_2, \dots, a_n \rangle$ называют такой элемент a_i , что $a_i = i$.

Узнав эти определения, он придумал собственную перестановку и назвал её *перестановкой Робина*.

Назовем перестановку $A = \langle a_1, a_2, \dots, a_{2n} \rangle$ из $2n$ натуральных чисел от 1 до $2n$ *перестановкой Робина*, если выполнены следующие условия:

- A имеет ровно n спусков, и все его спуски находятся на нечетных позициях (то есть $a_{2i-1} > a_{2i} < a_{2i+1}$ для всех i);
- A имеет ровно n неподвижных точек.

Например, перестановка $\langle 3, 2, 6, 4, 5, 1 \rangle$ является *перестановкой Робина*.

Кристофер решил поделиться своим открытием с Кроликом. Узнав о *перестановке Робина*, Кролик придумал следующее преобразование: удалим все неподвижные точки в последовательности и превратим оставшийся вектор в перестановку, заменив оставшиеся числа на количество элементов, не превосходящих его в перестановке. Например, преобразование перестановки $\langle 3, 2, 6, 4, 5, 1 \rangle$ дает $\langle 3, 2, 6, 4, 5, 1 \rangle \rightarrow \langle 3, 6, 1 \rangle \rightarrow \langle 2, 3, 1 \rangle$.

Кристофер теперь хочет получить по преобразованной перестановке *перестановку Робина*.

Формат входного файла

В первой строке входного файла дано n — число элементов в преобразованной перестановке ($1 \leq n \leq 100\,000$). Во второй строке входного файла дано n натуральных чисел — преобразованная перестановка.

Формат выходного файла

Если нет решения, вывести -1 в первой строке выходного файла. Если существует *перестановка Робина*, то вывести её. Если несколько решений, то вывести любую *перестановку Робина*.

Примеры

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

Задача G. Лесопилка

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

После того, как дуб, в котором жила Сова, упал, ей понадобилось новое жилище. Так как больше деревьев с большими удобными дуплами не осталось, Сова решила распилить несколько старых засохших деревьев на бревна.

Так как пилить Сове совсем не хочется, она обратилась на ближайшую лесопилку. Эта лесопилка, как и многие другие, использует инновационные технологии — на ней установлен автоматический разрезатель бревен. Он состоит из очень длинной линейки, вдоль которой отсчитываются координаты, фиксатора ствола дерева и n неподвижных лазерных пил. Левый конец разрезаемого ствола закрепляется фиксатором в некоторой точке, и ствол разрезается во всех местах, координаты которых совпадают с координатами пил.

Так как Сове для постройки дома подходят только бревна, длина которых не меньше a и не больше b , то для каждого из стволов деревьев, которые у нее есть, Сова хочет узнать, можно ли его распилить на лесопилке на такие бревна.

Формат входного файла

В первой строке входного файла записано одно целое число n ($1 \leq n \leq 10^6$) — количество пил на лесопилке. В следующей строке записано n целых чисел x_i ($|x_i| \leq 10^9$) — координаты i пилы. Координаты всех пил различны.

В следующей строке записано три целых числа: l ($1 \leq l \leq 10^9$) — длина ствола, который хочет распилить Сова, a и b ($1 \leq a \leq b \leq 10^9$) — длины самого короткого и самого длинного бревна, которые подходят Сове.

Формат выходного файла

В выходной файл выведите одно целое число — координаты точки, в которой нужно установить левый конец бревна, чтобы оно распилилось на подходящие Сове бревна. Если такой точки не существует, выведите «No solution».

Примеры

<code>sawmill.in</code>	<code>sawmill.out</code>
3 0 1 3 3 1 1	-1
3 0 1 3 4 1 1	No solution

Задача Н. Битва за мёд

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

Несколько дней Винни Пух он как ни в чём не бывало прогуливался по лесу, и попал в место, где раньше ни разу не был. Пух попал на клетчатую поляну, в каждой клетке которой находился улей с пчёлами. На несколько минут Винни замер и потерял дар речи. Ему казалось, что он попал в рай. Но что-то здесь было не так. . .

Заборы. Загородки. Ограды. Возможно, вы не понимаете удивления Пуха. Чтобы пояснить его, дадим небольшую справку. Пчёлы, как известно — очень преданные насекомые. Если они привязались к кому-то, то они будут очень долго служить только ему. Улей же, окружённый забором, является большой честью для пчёл. Так можно показать пчёлам свои добрые намерения, желание защищать их. Ведь никто не полезет через забор, а при отсутствии оно, к пчёлам может прийти кто угодно. Пчёлы это прекрасно понимают, и начинают служить тому, кто этот забор вокруг них возвёл.

Вернёмся же теперь к Винни. Заметив заборы, он сразу заподозрил что-то неладное. Они были явно искусственного происхождения. Действительно, откуда ещё могут взяться заборы посреди леса? Поняв, что у него есть конкурент и оценив, сколько мёда он потеряет, если ничего не предпримет, он решил действовать.

Пух затаился в лесу и стал ждать того незнакомца, который захотел заполучить себе местных пчёл. Всю ночь он провёл в его ожидании, но утром удача ему улыбнулась. Он услышал какой-то шум с противоположной стороны поляны и увидел незнакомца, который тащил доски с явным намерением использовать их в своих корыстных целях. К его несчастью, Винни тоже хорошо подготовился. А именно, в этот момент он уже бежал к выбранному улью, уже огороженному с трёх сторон, чтобы поставить четвёртое ограждение, тем самым переманив пчёл этого улья на свою сторону и захватив улей себе в подчинение.

Новообращенные пчёлы решили проявить перданность своему новому господину и окружили противника Пуха. Винни оценил, что пока его визави бегаёт от пчел из одного улья, у него самого есть время поставить еще одну перегородку и, возможно, захватить еще один улей. Пчелы из новозахваченного улья также нападают на противника Винни. К тому же так как отношения между пчелами из разных ульев не слишком дружелюбны, то если Пух, поставив одну доску, захватывает сразу два улья, то на его противника нападает только один улей. Пух понимает, что долго таким образом задерживать своего противника не сможет, и хочет узнать, сколько ульев он успеет захватить.

Формат входного файла

Во входном файле задана конфигурация поля. В первой строке заданы числа n и m ($1 \leq n, m \leq 1000$) — размеры поля. Вне поля ульев нет. Во второй строке задано число ульев t , рядом с которыми уже стоят заборы.

Далее в t строках идёт само их описание. Оно имеет вид $x_i y_i c_i \dots$ ($1 \leq x_i \leq n$, $1 \leq y_i \leq m$, $1 \leq c_i \leq 4$) — координаты улья, рядом с которым что-то есть, количество оград вокруг него, а затем, вместо \dots через пробел идут c_i различных букв N, S, E, W — с каких сторон улей окружён. N — север, S — юг, E — восток, W — запад. Ось x направлена на восток, y — на север.

Каждая перегородка является соседней для двух ульев. Если, например, у улья (x, y) на северной его части есть забор, то у улья с координатами $(x + 1, y)$ будет забор в южной части.

Во входном файле каждый кусок забора может быть указан один или два раза. То есть, если сказано, что у улья (x, y) на севере есть забор, то улей $(x + 1, y)$ может быть не указан во входном файле.

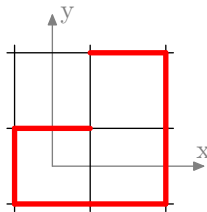
Формат выходного файла

Выведите единственное целое число — максимальное количество ульев, которые Винни успеет

окружить забором со всех сторон.

Примеры

honeywar.in	honeywar.out
2 2 3 1 1 3 W N S 2 1 2 S E 2 2 2 N E	3
1000 1000 3 3 3 3 N S W 4 3 3 N S E 8 8 4 N S W E	2



Задача I. Кто длиннее?

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

Как-то раз Тигра весело прыгал по лесу и придумывал новую игру. Вообще, как вам известно, Тигра очень любит различные игры, и чаще всего подвижные. Но в тот день над лесом постепенно сгущались тучи, и потому все мысли Тигры были о тихой и спокойной игре.

И вот тут Тигра вспомнил о очень веселой и занимательной игре, в которую они с Кроликом играли в далеком детстве. Правила ее хоть и не были просты, но игра занимала большое количество времени и приносила массу удовольствий.

Через пару минут Тигра встретил ослика Иа-Иа и предложил ему сыграть. Ослик согласился и тогда Тигра поведал ему правила игры: игроки по очереди приписывают к строчке одну из известных им букв английского алфавита, но при этом в строке не должно образоваться двух одинаковых комбинаций из двух подряд идущих символов. Тот, кто не может сделать ход — проигрывает.

Иа-Иа очень заинтересовался этой игрой и захотел выиграть у Тигры. Для начала он хочет придумать наидлиннейшую строку, которая может быть получена в процессе этой игры. Помогите ему в этом!

Формат входного файла

Единственная строка входного файла содержит число k ($1 \leq k \leq 26$) — количество первых букв английского алфавита, известное героям.

Формат выходного файла

В единственной строке файла должна быть строка, состоящая из k первых строчных букв английского алфавита — ответ на задачу.

Примеры

	<code>string.in</code>	<code>string.out</code>
1		aa

Задача J. Часы

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

На очередной день рождения ослику Иа-Иа подарили наручные стрелочные часы. Теперь у него появилось новое развлечение — смотреть на бег стрелок. На то, как минутная догоняет часовую, обходит и тут же продолжает бежать за ней.

Вот и в этот раз Кенга застала ослика за этим занятием. Она присоединилась к наблюдением и через некоторое время ей стало интересно, сколько уже моментов, когда минутная стрелка обгоняет часовую, видел Иа-Иа. Для этого она спросила у ослика во сколько он начал смотреть на часы, записала это и текущее время и побежала к Сове с этим вопросом. Но Сова оказалось очень занята и поэтому попросила вас помочь.

Как известно, за один день часовая стрелка делает два оборота, а минутная целых 24.

Формат входного файла

Первая строка входного файла содержит описание момента времени, в который Иа-Иа начал смотреть на часы в формате НН:ММ — две первых цифры обозначают часы, потом через двоеточие две цифры минут.

Вторая строка входного файла содержит описание конечного момента времени в таком же формате.

Известно, что Иа-Иа наблюдал за часами менее суток, но вполне мог это делать и в полночь.

Формат выходного файла

В выходной файл выведите единственное целое число — количество моментов, когда минутная стрелка обгоняет часовую в процессе наблюдения.

Если такое событие происходило когда Иа-Иа начал или закончил смотреть на часы, то оно также считается.

Примеры

<code>watches.in</code>	<code>watches.out</code>
23:50 00:20	1
12:00 13:20	2