

Задача А. Считаем конфеты

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вам дана матрица с n строками и m столбцами. Эта матрица представляет собой вид сверху коробки конфет. Матрица будет содержать только следующие символы:

- “.” обозначает свободное место
- “o” обозначает съедобную часть конфеты
- “<>v^” обозначает фантик

Получается, есть ровно два способа, как выглядит целая конфета:

```
.....V.  
>o<..o.  
.....^.
```

Всякий раз, когда вы видите три символа, расположенных таким образом, вы видите целую конфету. Считайте, что ситуация, как на картинке ниже, невозможна

```
.v.  
>o<  
.^.
```

Посчитайте количество целых конфет, которые вы видите.

Формат входных данных

В первой строке даны два целых числа n и m ($1 \leq n, m \leq 400$). Следующие n строк содержат по m символов из условия выше, задающих коробку конфет.

Формат выходных данных

Выведите одно число — ответ на задачу.

Пример

стандартный ввод	стандартный вывод
5 4 . >o< v. ^. ooo. ^. ^. >o<<	3

Задача В. Дерево и бензобаки

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В стране есть n городов, соединённых сетью из $n - 1$ двусторонних дорог, образующих дерево. Демиду нужно посетить каждый из этих городов, и он уже определил порядок своих визитов. Из каждого города в следующий пункт его маршрута он будет ездить на арендованном в этом городе автомобиле. Каждая аренда совершенно бесплатна, но автомобили необходимо заправлять – машина с баком ёмкостью k должна быть заправлена в самом начале, а затем заправляться после каждых k пройденных дорог. В конечном городе маршрута также нужно заправиться, поскольку машину нужно возвращать с полным баком.

Зная маршрут героя, цены на топливо и ёмкости баков, определите, сколько будет стоить каждый отрезок маршрута. Стоимость заправки полного бака топлива в городе i равна c_i и не зависит от размера бака.

Формат входных данных

В первой строке дано целое число n ($2 \leq n \leq 5 \cdot 10^4$). Города пронумерованы от 1 до n . Во второй строке содержится последовательность из n целых чисел c_1, \dots, c_n ($1 \leq c_i \leq 10^4$). Следующие $n - 1$ строки описывают дорожную сеть. Каждая из этих строк содержит два целых числа u и v ($1 \leq u \neq v \leq n$), которые указывают на то, что между городами под номерами u и v есть двухсторонняя дорога.

После этого следует строка с последовательностью из n целых чисел t_1, \dots, t_n , которые задают порядок посещения городов Демидом (каждое из чисел от 1 до n встречается ровно один раз в этой последовательности). Последняя строка ввода содержит последовательность из $n - 1$ целых чисел k_1, \dots, k_{n-1} , которые указывают ёмкости баков арендованных автомобилей: число k_i указывает на то, что во время поездки из города t_i в город t_{i+1} нужно заправить свой автомобиль после каждых k_i пройденных дорог. Гарантируется, что k_i всегда делит расстояние между этими городами.

Формат выходных данных

Выведите $n - 1$ число — i -е из них должно быть стоимостью доехать от города t_i до города t_{i+1} .

Пример

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

Задача С. Теперь про время

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В галактике X используют две системы обозначения времени. В каждой системе k -часовой день делится на несколько отрезков различной длины, и в каждом отрезке часы отсчитываются независимо.

В системе А день делится на n отрезков. Первый отрезок содержит a_1 часов и обозначается АА, второй отрезок содержит a_2 часов и обозначается АВ, ..., n -й отрезок содержит a_n часов и обозначается А\$, где \$ — n -й символ латинского алфавита.

В системе В день делится на m отрезков. Первый отрезок содержит b_1 часов и обозначается ВА, второй отрезок содержит b_2 часов и обозначается ВВ, ..., m -й отрезок содержит b_m часов и обозначается В#, где # — m -й символ латинского алфавита.

Однако есть небольшая проблема. На самом деле все не так просто. В каждой системе день действительно делится на отрезки соответствующей длины, но порядок отрезков в каждом дне не фиксирован. Так, например, в один из дней отрезки в системе А могут идти в порядке 3, 2, 1, а отрезки в системе В — в порядке 1, 4, 2, 3 (если в этих системах 3 и 4 отрезка, соответственно).

Однажды к вам в руки попал дневник жителя этой галактики. Он делал отметки о различных интересных моментах времени в обеих временных системах. Так что у вас есть некоторая информация о соответствии времен в системах. Можете ли вы выяснить порядок следования отрезков в обеих системах в этот день?

Формат входных данных

Первая строка содержит k ($1 \leq k \leq 99$). Вторая строка содержит число n , после которого следуют числа a_1, a_2, \dots, a_n ($1 \leq n \leq 8$, $1 \leq a_i \leq k$, сумма всех a_i равна k). Третья строка содержит число m , после которого следуют числа b_1, b_2, \dots, b_m ($1 \leq m \leq 8$, $1 \leq b_i \leq k$, сумма всех b_i равна k).

Четвертая строка содержит t — количество известных соотношений между временами в обеих системах ($0 \leq t \leq 100$). Следующие t строк содержат соотношения. В каждом соотношении времена заданы как ХХ:УУСС, где ХХ — количество часов, УУ — количество минут, а СС означает название отрезка. ХХ изменяется от 00 до 98 (к счастью, жители планеты используют 00:00АА, 00:01АА и т.д. для обозначения времени в первом часе отрезка), УУ изменяется от 00 до 59, СС изменяется от АА до А\$ или от ВА до В#.

Формат выходных данных

Если вы не можете восстановить информацию о порядке следования отрезков в дне в обеих системах из-за неоднозначности, выведите “Ambiguous”. Если не существует ни одного способа восстановить порядок следования отрезков, выведите “Inconsistent”. В противном случае выведите порядок следования отрезков в каждой из систем, следуя формату, приведенному в примере.

Примеры

стандартный ввод	стандартный вывод
24 2 14 10 2 12 12 1 05:31AA=03:31BA	A: 2 1 B: 2 1
24 2 12 12 2 12 12 1 07:39AA=07:39BA	Ambiguous
24 2 12 12 2 12 12 1 07:39AA=07:38BA	Inconsistent

Задача D. Самоописывающие числа

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Число 31143310 является самоописывающим, потому что мы можем прочесть его как утверждение «это число содержит три 1, одну 4, три 3 и один 0», и это утверждение правдиво для этого числа.

Более формально, число называется самоописывающим, если оно удовлетворяет следующим условиям:

- Оно имеет четное количество цифр. Обозначим отдельные цифры как $a_0, b_0, a_1, b_1, \dots$ слева направо.
- Все цифры b_i различны.
- Для каждого допустимого i число содержит ровно a_i копий цифры b_i .
- Число не содержит никаких других цифр, кроме тех, которые описаны в утверждениях, упомянутых выше.

Вам дано множество цифр. Найдите наименьшее самоописывающее число x , которое состоит только из цифр множества, и каждая цифра множества встречается хотя бы один раз.

Формат входных данных

Дана строка, состоящая из различных цифр.

Формат выходных данных

Если ответ существует, выведите его, иначе выведите -1 .

Примеры

стандартный ввод	стандартный вывод
0134	10143133
1	-1

Задача Е. Соединяем точки с землей

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

У вас есть $2n$ точек на плоскости двух типов. Точки первого типа имеют координаты $(c_i, 0)$, то есть $y = 0$. Точки второго типа имеют координаты (x_j, y_j) , причем $y_j > 0$. Точек каждого типа одинаковое количество. Нужно посчитать число способов разбить точки на пары (в паре две точки разного типа, каждая точка ровно в одной паре) так, чтобы отрезки, соединяющие точки в паре, не пересекались (даже в концах отрезков).

Формат входных данных

В первой строке дано целое число n ($1 \leq n \leq 50$). Вторая строка содержит n целых чисел c_i ($1 \leq c_i \leq 5000$). Далее идут n строк, содержащих по два целых числа x_j, y_j ($1 \leq x_j, y_j \leq 5000$). Все $2n$ точек различны.

Формат выходных данных

Выведите одно число — ответ на задачу по модулю $10^9 + 7$.

Пример

стандартный ввод	стандартный вывод
3 3 20 1 1 5 2 2 3 10	2

Задача F. Треугольники на плоскости

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Перед вами белая плоскость. У вас есть n прямоугольных равнобедренных треугольников, которые вы последовательно применяете к плоскости. Каждый треугольник задаётся числами x, y, r , которые задают вершины треугольника (x, y) , $(x + r, y)$ и $(x, y + r)$. Треугольник применяется следующим образом — вы инвертируете цвет плоскости под ним, с белого на черный и с черного на белый. Нужно посчитать суммарную площадь черной части после применения всех треугольников.

Формат входных данных

В первой строке дано целое число n ($1 \leq n \leq 10$). Следующие n строк содержат по три целых числа x, y, r ($1 \leq x, y, r \leq 10^6$).

Формат выходных данных

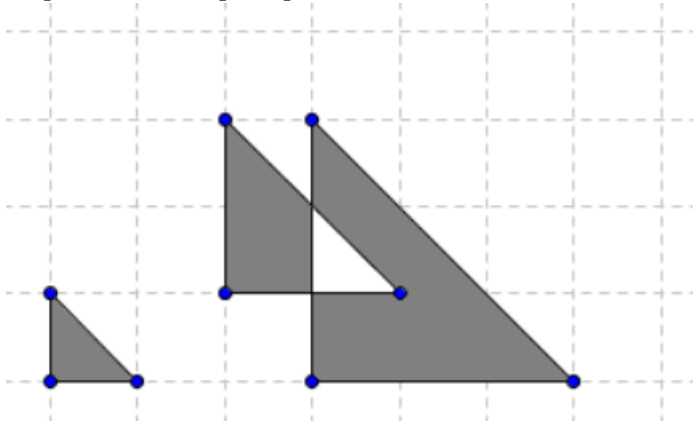
Выведите одно число — ответ на задачу, с абсолютной точностью.

Пример

стандартный ввод	стандартный вывод
3 1 1 1 3 2 2 4 1 3	6.0

Замечание

Картинка для примера



Задача G. Восстановить значения в дереве

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вам дано подвешенное дерево из n вершин. Корнем дерева является вершина 1. Исходно в дереве в каждом листе было записано некоторое положительное целое число, а в остальных вершинах была записана сумма чисел во всех детях. Некоторые из чисел, записанных в вершинах дерева, потеряны. Восстановите изначальные числа, если это можно сделать единственным образом, или сообщите, что это невозможно.

Формат входных данных

Первая строка содержит целое число n ($2 \leq n \leq 3 \cdot 10^5$). Следующая строка содержит $n - 1$ число p_2, p_3, \dots, p_n ($1 \leq p_i < i$), где p_i это номер вершины, являющейся предком вершины i . Следующая строка содержит n целых чисел a_i ($0 \leq a_i \leq 10^9$) — числа, записанные в вершине i . Если $a_i = 0$, то число потеряно. Ограничение на a_i накладывается только на входные числа, потерянные числа могут быть больше.

Формат выходных данных

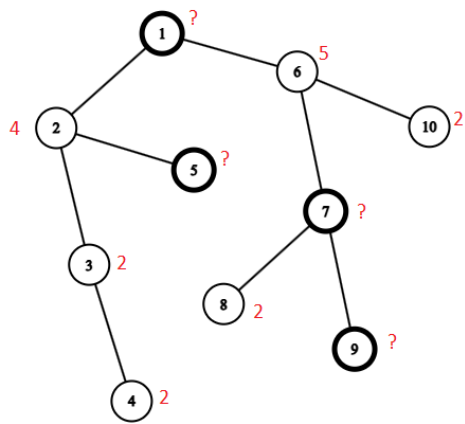
Если числа нельзя восстановить или числа восстанавливаются не единственным способом, то выведите impossible. Иначе выведите n положительных целых чисел x_i — числа, записанные в вершине i .

Примеры

стандартный ввод	стандартный вывод
10 1 2 3 2 1 6 7 7 6 0 4 2 2 0 5 0 2 0 2	9 4 2 2 2 5 3 2 1 2
5 1 2 2 1 4 0 0 0 1	impossible

Замечание

Картинка с первым примером



Задача Н. Упаковываем числа

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

У вас есть числа от 1 до n , а именно a_i чисел i . Вы хотите сгруппировать эти числа в наборы по k чисел, удовлетворяя следующим условиям:

- В наборе должно быть хотя бы два разных числа.
- Если $a_i > 1$, то все копии числа i не должны находиться в одном наборе.

Эти условия могут сделать невозможным группировку чисел в наборы, например, при $n = 1$. При этом, если $\sum a_i$ не кратно k , то некоторые числа могут не войти ни в один набор, в этом нет ничего страшного. Чтобы создать как можно больше наборов, вы можете брать любое число и превращать его в любое другое (возможно, большее n) за одну операцию. Какое наибольшее число наборов чисел вы можете таким образом сгруппировать, и какое наименьшее число операций для этого нужно применить?

Формат входных данных

В первой строке даны два целых числа n и k ($1 \leq n \leq 10^5$, $2 \leq k \leq 10^9$). Вторая строка содержит n целых чисел a_i ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Сначала выведите наибольшее число наборов, которые можно сгруппировать, а затем наименьшее число операций, которые для этого нужно выполнить.

Примеры

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

Задача I. Взаимнопростое дерево на окружности

Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

На окружности написано n целых чисел. Вы можете соединять хордой два числа, если они не взаимнопростые. Ваша цель соединить некоторые числа так, чтобы получилось дерево, то есть по хордам можно было бы добраться от любого числа до любого, при этом хорды не должны пересекаться (кроме как в точках на окружности). Сколько существует способов добиться этой цели?

Два способа называются различными, если в одном способе a_i и a_j соединены хордой, а в другом нет.

Формат входных данных

Первая строка содержит одно целое число n ($2 \leq n \leq 300$). Каждая из следующих n строк содержит целое число a_i ($2 \leq a_i \leq 10^9$) — числа в порядке их записи на окружности.

Формат выходных данных

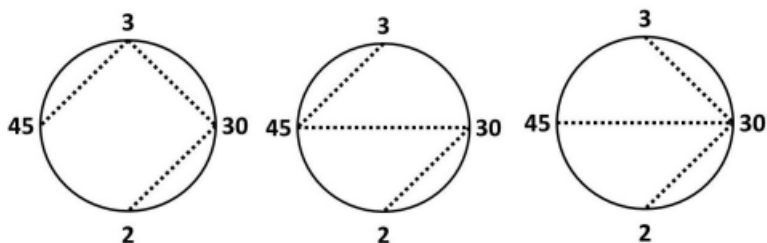
Выведите одно целое число — ответ на задачу. Поскольку это число может быть очень большим, выведите его по модулю $10^9 + 7$.

Пример

стандартный ввод	стандартный вывод
4 3 30 2 45	3

Замечание

В примере есть три способа



Задача J. Ценные точки

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

На числовой прямой обозначены n точек, каждая из которых имеет координату x_i и ценность c_i . Вы должны выбрать ровно m точек таким образом, чтобы между любыми выбранными точками расстояние было хотя бы d , а минимальная ценность среди выбранных точек была наибольшей.

Формат входных данных

Первая строка содержит три целых числа n, m, d ($1 \leq m \leq n \leq 10^5$, $1 \leq d \leq 10^9$). Следующие n строк содержат по два целых числа x_i и c_i ($1 \leq x_i \leq 10^9$, $x_i \neq x_j$ при $i \neq j$, $1 \leq c_i \leq 10^9$).

Формат выходных данных

Выведите наибольшую минимальную ценность выбранных точек. Если выбрать требуемым образом точки невозможно, выведите -1 .

Пример

стандартный ввод	стандартный вывод
5 2 2 1 10 3 15 2 20 5 8 4 11	11

Задача К. Всё битовое не ноль

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вам дан массив a длины n . Вы можете за одну операцию увеличить любой элемент на один, либо уменьшить на один (уменьшать ноль запрещено). Операцию можно применять к одному числу несколько раз. За наименьшее число операций нужно добиться того, чтобы побитовый OR, AND и XOR были больше нуля.

Формат входных данных

В первой строке дано целое число n ($2 \leq n \leq 5 \cdot 10^5$). Вторая строка содержит n целых чисел a_i ($0 \leq a_i \leq 10^{18}$).

Формат выходных данных

Выведите одно число — ответ на задачу.

Пример

стандартный ввод	стандартный вывод
4 5 2 3 3	1

Задача L. Пирамидизация

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Назовем последовательность чисел a_1, a_2, \dots, a_k пирамидой, если существует j ($1 \leq j \leq k$), такое, что $a_1 \leq a_2 \leq \dots \leq a_j$ и $a_j \geq a_{j+1} \geq \dots \geq a_k$.

Вы можете прибавлять один к элементу массива за одну операцию. Стоимостью пирамидизации последовательности назовем минимальное количество операций, которые нужно применить, чтобы последовательность стала пирамидой.

Вам дана последовательность из n чисел a_1, a_2, \dots, a_n . Обозначим за $f(i, j)$ стоимость пирамидизации последовательности a_i, a_{i+1}, \dots, a_j . Нужно посчитать суммарную стоимость пирамидизации всех подотрезков исходной последовательности, то есть значение

$$\sum_{1 \leq i \leq j \leq n} f(i, j)$$

Формат входных данных

В первой строке дано целое число n ($1 \leq n \leq 10^6$). Вторая строка содержит n целых чисел a_i ($1 \leq a_i \leq 10^9$).

Формат выходных данных

Выведите одно число, ответ на задачу, по модулю $10^9 + 7$.

Пример

стандартный ввод	стандартный вывод
4 2 1 1 2	2

Задача М. Удаляем различные

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Вам дан набор из n целых чисел a_1, a_2, \dots, a_n . За одну операцию вы можете удалить k различных чисел из набора. Для каждого k от 1 до n определите, какое наибольшее количество раз вы можете проделать такую операцию с набором?

Формат входных данных

Первая строка содержит целое число n ($1 \leq n \leq 3 \cdot 10^5$). Вторая строка содержит n целых чисел a_i ($1 \leq a_i \leq n$).

Формат выходных данных

Выведите n чисел, i из этих чисел должно быть ответом на задачу при $k = i$.

Пример

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

Замечание

В примере при $k = 2$ мы можем сделать только одну операцию, потому что после неё останутся два одинаковых числа.

Задача N. Отрезки и окружность

Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дано n точек на плоскости, а также окружность с центром в точке $(0, 0)$ и радиусом r . Все точки находятся вне окружности. Необходимо определить, сколько отрезков, концами которых являются заданные точки, не пересекаются с окружностью.

Формат входных данных

В первой строке даны два целых числа n и r ($1 \leq n \leq 5 \cdot 10^4$, $1 \leq r \leq 10^6$). Следующие n строк содержат по два целых числа x_i, y_i ($-10^6 \leq x_i, y_i \leq 10^6$). Гарантируется, что никакая прямая, проходящая через пару данных точек, не является касательной к окружности.

Формат выходных данных

Выведите одно число — ответ на задачу.

Пример

стандартный ввод	стандартный вывод
4 6 0 9 0 -9 9 0 -9 0	4