

## [С практика] 1. Вика и Витя

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

Вика и Витя играют в любопытную игру. Вика выкладывает вслепую на стол  $n$  карточек, на каждой  $i$ -й из них написано число  $a_i$ . Перед этим Витя загадывает некоторое число  $k$ . Если сумма чисел на каких-то  $n - 1$  карточках будет хотя бы  $k$ , то выиграл Витя. Иначе – Вика. Определите, кто из ребят победит.

### Формат входных данных

В первой строке записано натуральное число  $k$  ( $1 \leq k \leq 10^9$ ). Во второй строке записано натуральное число  $n$  ( $1 \leq n \leq 1000$ ). В третьей строке даны  $n$  натуральных чисел  $a_i$  через пробел ( $1 \leq a_i \leq 3000$ ).

### Формат выходных данных

Если выиграет Витя, то выведите «YES», иначе – «NO».

### Примеры

стандартный ввод	стандартный вывод
2021 4 500 1200 100 250	NO
2020 2 3000 1000	YES

### Замечание

В первом примере в любой тройке сумма чисел будет менее 2021.

- $500 + 1200 + 100 = 1800$
- $500 + 1200 + 250 = 1950$
- $500 + 100 + 250 = 850$
- $1200 + 100 + 250 = 1550$

Во втором примере достаточно убрать из двух чисел число 1000 – останется 3000, сумма оставшихся чисел хотя бы 2020.

## [С практика] 2. Инициатива

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

Специалисты службы информационной безопасности наняли стажера Васю. Вася захотел проявить себя в первый же день и придумал собственный шифр. Произвольная строка  $S$  обрабатывается следующим образом: буквы строки  $S$  переставляются, символы с начала строки и с конца станут чередоваться. Первый символ строки  $S$  будет первым символом зашифрованной строки. Последний символ станет вторым, затем второй символ попадёт на третью позицию, а предпоследний – на четвертую и так далее, пока в строке останутся символы. Затем к полученной строке приписывается справа служебный символ  $\#$ .

Вася случайно запустил шифрующий скрипт на базе данных клиентов и ушёл в долгосрочный отпуск. Поэтому расшифровывать базу придётся вам.

### Формат входных данных

В единственной строке дана зашифрованная строка из больших и маленьких латинских букв и символа  $\#$ . Гарантируется, что данная строка могла быть получена применением описанного выше алгоритма. Длина строки не превосходит 100 символов.

### Формат выходных данных

Выведите единственную строку – расшифрованный текст.

### Пример

стандартный ввод	стандартный вывод
Tfifnok#	Tinkoff

### Замечание

Строка Tinkoff зашифровывалась бы следующим образом:

- T – добавили букву с первой позиции
- Tf – добавили букву с 7-й позиции
- Tfi – добавили букву со 2-й позиции
- Tfif – добавили букву с 6-й позиции
- Tfifn – добавили букву с 3-й позиции
- Tfifno – добавили букву с 5-й позиции
- Tfifnok – добавили букву с 4-й позиции
- Tfifnok# – дописали символ  $\#$

Результат : Tfifnok#

## [С практика] 3. Конь Юлий

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

Юлий — особый шахматный конь. Каждый год он делает ровно один ход на бесконечной шахматной доске. По високосным годам он ходит на две клетки вверх и на одну вправо, а по обычным — на одну вверх и две вправо.

Изначально Юлий находится в клетке с координатами  $(0, 0)$ . Где он окажется, после совершения  $n$  ходов, начиная с года  $x$ ?

### Формат входных данных

В первой строке дано число  $x$  ( $1 \leq x \leq 10^9$ ) — первый год, в который Юлий сделает ход.  
Во второй строке дано число  $n$  ( $1 \leq n \leq 10^9$ ) — количество ходов.

### Формат выходных данных

Выведите координаты клетки, в которой окажется Юлий после  $n$  ходов.

### Примеры

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

### Замечание

Напомним, что год является високосным, если он кратен 400, либо кратен 4 и не кратен 100.

## [С практика] 4. Москва-Владивосток

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

Берляндия славится своими поездами. Вот и вы захотели почувствовать романтику плацкарта и пустились в незабываемое путешествие на самом новом скоростном поезде. Но вот незадача: оказывается, через всю Берляндию проходит лишь одна железная дорога (то есть железными дорогами Берляндия не славится).

В данный момент по железной дороге один за другим едут  $n$  поездов. Максимальная скорость  $i$ -го поезда —  $a_i$  км/ч. Будем считать, что  $(i+1)$ -й поезд едет за  $i$ -м. Разумеется, поезда не могут обгонять друг друга. Для каждого поезда выясните, с какой скоростью он может ехать в сложившейся ситуации.

### Формат входных данных

В первой строке дано число  $n$  ( $1 \leq n \leq 10^5$ ) — количество поездов. Далее в  $n$  строках вводится по одному целому числу  $a_i$  ( $1 \leq a_i \leq 10^9$ ) — скорость  $i$ -го поезда.

### Формат выходных данных

Выведите  $n$  чисел — ответ для каждого поезда.

### Пример

стандартный ввод	стандартный вывод
3 100 110 60	100 100 60

## [С теория] 1. Я хочу сыграть с тобой в игру

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

Глеб любит играть в настолки. Но еще больше Глеб любит их придумывать. В очередной раз он предложил опробовать его новое творение. На этот раз игра предназначена для одного человека.

В начале вам выдается 6 карточек с цифрами от 1 до 6 (каждая цифра встречается ровно один раз). Затем в каждый свой ход вы можете заменить ровно 2 свои карточки на «соседние» по номиналу. То есть карточку с цифрой 2 можно заменить на карточку с цифрой 1 или 3, карточку с цифрой 3 — на 2 или 4. Причем карточку с цифрой 1 можно поменять не только на 2, но и на 6. А карточку с цифрой 6 можно заменить на карточку с цифрой 1 или 5.

Цель игры — получить 6 карточек с одинаковыми цифрами. Выясните, получится ли у вас выиграть и, если получится, предъявите последовательность ходов.

## [С теория] 2. Мафия

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

Вы внедрили в мафиозную группировку под прикрытием. Разумеется, боссу здесь подчиняются все, а босс никому не подчиняется.

В группировке  $n$  человек, **не считая вас**. Вы можете проверить, подчиняется ли человек  $i$  человеку  $j$ . Сможете ли вы определить босса мафии, сделав менее  $n$  проверок? Если да, то как? Если нет, докажите, почему.

Так как группировка работает в Берляндии, то  $A$  может подчиняться  $B$ , а  $B$  подчиняться  $A$  одновременно.

Так же стоит отметить, что вы, к сожалению, не босс.

## [C-В' практика] 1. Создаем игры

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

В последнее время достаточно популярной механикой в играх становится управление гравитацией. Глеб решил не оставаться в стороне и также создать свою игру про управление гравитацией. В начале всегда нужно сделать миниверсию игры, поэтому он решил что самой базовой фишкой его игры будет переворот экрана на 90 градусов по часовой стрелке. Глеб решил сделать игру про кубики. Изначально у вас есть коробка  $n$  на  $n$ , в которой лежат кубики, в  $i$ -ой стопке  $a_i$  кубиков. Ваша задача сказать, как будут выглядеть стопки кубиков после  $k$  поворотов на 90 градусов по часовой стрелке. Так как игра про гравитацию, то после каждого поворота кубики падают (более подробно как работает падение смотрите в пояснении).

### Формат входных данных

В первой строке даются два числа  $n, k$  ( $1 \leq n \leq 10, 0 \leq k \leq 10^9$ ) — количество стопок и поворотов соответственно. В следующей строке даются  $n$  чисел:  $i$ -е число обозначает  $a_i$  ( $0 \leq a_i \leq n$ ).

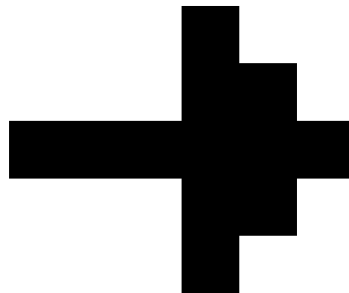
### Формат выходных данных

Выведите  $n$  чисел  $b_i$  — количество элементов в  $i$ -й стопке после  $k$  итераций.

### Пример

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

### Замечание



## [С-В' практика] 2. Гурман

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

Нитуширг в жизни больше всего любит две вещи - вкусно поесть и отдыхать на море. Поэтому он решил совместить приятное с приятным и поехать на пятизвёздочный морской курорт Фокьнит. Но вот незадача: в Фокьните есть только одна столовая, меню в которой каждый день заранее фиксировано. Поскольку Нитуширг - гурман, он предпочитает разнообразие в еде, поэтому если за время отдыха какое-то меню попадётся более  $K$  раз - Нитуширг очень расстроится и весь отпуск пойдёт насмарку. При этом, разумеется, Нитуширг хочет отдохнуть на море как можно дольше.

Ваша задача состоит в том, чтобы по известному заранее меню столовой на  $N$  дней выбрать как можно более длинный отрезок подряд идущих дней, в которые Нитуширг поедет в Фокьнит так, чтобы никакой набор блюд не повторялся более  $K$  раз за всё время поездки.

### Формат входных данных

В первой строке вам даны целые числа  $1 \leq N \leq 10^5$  и  $1 \leq K \leq N$  - количество дней, когда Фокьнит открыт для гостей и  $K$  - максимальное количество раз, которое может повториться меню за время отпуска. В следующей строке входного файла вам дана строка, состоящая из  $N$  строчных латинских букв. Каждая буква означает свой вариант меню, разным меню соответствуют разные буквы.

### Формат выходных данных

На выход ваша программа должна вывести два целых числа:  $1 \leq L \leq N$  - максимальное количество дней отпуска Нитуширга и  $1 \leq d_0 \leq N$  - номер дня заезда. Считайте, что в день заезда Нитуширг успевает съесть предлагаемое столовой меню.

Если подходящих  $d_0$  несколько - выведете наименьшее из них: Нитуширг любит отдыхать в начале лета.

### Примеры

стандартный ввод	стандартный вывод
1 1 a	1 1
6 2 abbbaa	4 3



## [С-В' практика] 3. Физрук Арсений

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

В начале урока физкультуры ученики 13А класса выстроились в ряд. Физрук Арсений любит порядок, но школьники опять встали не по росту. Он решил проучить их и выбрать какой-то *хороший отрезок* детей, и отправить их играть в волейбол, а остальных оставить выполнять нормативы. *Хорошим отрезком* детей Арсений называет такой непрерывный отрезок детей в ряду, что их рост строго убывает.

Ученики любят волейбол, поэтому хотят понять, есть ли у них шанс оказаться в числе счастливых. Для этого каждый школьник хочет выяснить, как много людей может пойти играть с ним в волейбол, то есть найти длину наибольшего *хорошего отрезка*, содержащего его самого.

### Формат входных данных

Первая строка содержит одно целое число  $n$  ( $1 \leq n \leq 10^5$ ) — количество учеников 13А, пришедших на урок.

Вторая строка содержит  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ) — рост школьников в том порядке, в котором они встали изначально.

### Формат выходных данных

Выведите  $n$  целых чисел через пробел, где  $i$ -е число — максимальная длина *хорошего отрезка*, содержащего школьника номер  $i$ .

### Примеры

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

### Замечание

Заметьте, что в первом примере рост школьников в отрезке с первого по четвертого (7 4 2 2) не является строго убывающим

## [С-В' практика] 4. Агата и Кристи

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	256 мегабайт

Кристи и Агата – лучшие подруги. Они любят проводить свободное время за разгадыванием головоломок. В этот раз загадку подготовила Кристи, помогите Агате справиться с задачей.

Даны  $n$  чисел  $a_i$  ( $1 \leq a_i \leq 2 \cdot 10^5$ ). Также имеются  $q$  запросов вида  $(l_j, r_j)$  – такой запрос означает, что надо сложить все числа на отрезке от  $l_j$  до  $r_j$  включительно и вернуть полученную сумму. Кристи дала Агате набор чисел и сами запросы. Задача Агаты – расставить числа  $a_i$  на позициях от 1 до  $n$  (по одному на позицию) так, чтобы сумма ответов на все  $q$  запросов была как можно больше.

Помогите Агате!

### Формат входных данных

В первой строке через пробел даны количество чисел в наборе –  $n$  ( $1 \leq n \leq 2 \cdot 10^5$ ) и количество запросов  $q$  ( $1 \leq q \leq 2 \cdot 10^5$ ).

Во второй строке даны числа  $a_i$  ( $1 \leq a_i \leq 2 \cdot 10^5$ ), записанные через пробел.

В следующих  $q$  строках даны запросы по одному в строке. Запрос  $j$  состоит из двух чисел  $l_j$  и  $r_j$ , записанных через пробел ( $1 \leq l_j \leq r_j \leq n$ ).

### Формат выходных данных

Выведите единственное число – максимальную сумму ответов на запросы, которую может получить Агата.

### Примеры

стандартный ввод	стандартный вывод
5 3 1 9 4 3 1 1 4 2 2 3 5	34
1 1 5 1 1	5
2 2 8 4 1 1 2 2	12

### Замечание

Рассмотрим первый пример: выгоднее всего будет поставить 9 на позицию 4, а 4. на позицию 3, а 3 на позицию 2, единицы поставим на первую и последнюю позиции. Получим: 1 3 4 9 1.

Какие результаты вернут запросы суммы:

- $(1, 4) \rightarrow 17$
- $(2, 2) \rightarrow 3$
- $(3, 5) \rightarrow 14$

Сумма всех ответов равна 34.

## [С-В' теория] 1. Суслики и панды

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

33 Суслика и 33 панды решили пообедать вместе. Все животные сели за круглый стол. Докажите, что у кого-то из животных оба соседа – суслики.

## [C-V' теория] 2. А и Боба

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

А и Боба живут в городе  $N$  во Флатландии. Карта города представляет собой бесконечную квадратную сетку, в которой каждая улица идёт либо по горизонтали, либо по вертикали. Кроме того, расстояние между любыми двумя соседними горизонтальными улицами равно  $H$ , а между любыми двумя соседними вертикальными равно  $W$ .

Все улицы в городе  $N$  нумеруются от центра города, имеющего координаты  $(0, 0)$ , причём номерами улиц являются все целые числа и нумерация вертикальных улиц не зависит от нумерации горизонтальных. Например, в городе есть Первая вертикальная улица и Первая горизонтальная улица, которые пересекаются в перекрёстке с координатами  $(W, H)$ . А МинусПервая горизонтальная и МинусВторая вертикальная улица пересекаются в перекрёстке с координатами  $(-2W, -H)$ . Кроме того в каждой клетке прямоугольной сетки города  $N$  находится бесконечно высокий небоскрёб.

В один из дней А и Боба решили провести эксперимент: А встал на пересечении вертикальной улицы с номером  $a$  и горизонтальной улицы с номером  $b$ , а Боба встал на пересечении вертикальной улицы с номером  $c$  и горизонтальной улицы с номером  $d$ . Теперь они хотят посмотреть друг на друга, но есть проблема: между ними находятся небоскрёбы, которые заслоняют обзор. А и Боба хотят снести некоторое количество небоскрёбов так, чтобы оставшиеся не мешали им смотреть друг на друга, находясь в точках  $(a, b)$  и  $(c, d)$ . Считается, что небоскрёб заслоняет обзор, если отрезок между А и Бобой пересекает внутренность прямоугольника небоскрёба, и не заслоняет, если он пересекает границу небоскрёба или один из углов.

Помогите А и Бобе понять, какое минимальное количество небоскрёбов им придётся снести!

### Формат входных данных

Вам даны 6 целых чисел:  $W, H, a, b, c, d$ .

### Формат выходных данных

В решении вам нужно предъявить и доказать формулу, по которой можно посчитать минимальное количество небоскрёбов, которое нужно снести.

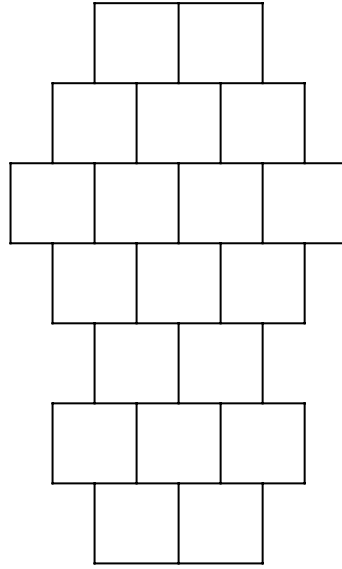
### Примеры

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

## [В'-В практика] 1. Стена

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

Вам дано  $n$  кубиков. Вам предлагается построить стену из  $n$  кубиков высотой в  $h$  слоёв. Основание стены будет состоять из  $b$  кубиков. Каждый следующий слой отличается по количеству кубиков ровно на 1. Пример корректной башни приведён на рисунке ниже. Она состоит из 19 кубиков, 7 слоёв, а размер её основания — 2.



Найдите количество способов сделать указанное (построить башню из  $n$  кубиков с размером основания в  $b$  кубиков и высотой в  $h$  слоёв). Так как ответ может быть очень большим, выведите его по модулю  $10^9 + 7$ .

### Формат входных данных

В единственной строке через пробел даются 3 числа —  $n$ ,  $b$  и  $h$  ( $1 \leq n \leq 1000$ ,  $1 \leq b, h, \leq 100$ ).

### Формат выходных данных

Выведите единственное число — число способов построить стенку, взятое по модулю  $10^9 + 7$ .

### Примеры

стандартный ввод	стандартный вывод
15 5 5	1
16 5 5	0
22 2 8	7
10 2 5	2

## [В'-В практика] 2. Жадные черепахи

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

$N$  жадных черепах, пронумерованных от 1 до  $N$  стоят в очереди перед автоматом, выдающим подарки. После того, как черепаха под номером  $i$  получает подарок, она снова встает в очередь так, что за ней оказывается ровно  $a_i$  черепах.

Можно заметить, что даже при бесконечном запасе подарков в автомате, какие-то черепахи все равно останутся без подарков. Определите, сколько черепах не получат ни одного подарка, если число подарков в автомате может быть сколь угодно велико.

### Формат входных данных

В первой строке содержится одно целое число  $N$  ( $1 \leq N \leq 10^5$ ) — число черепах. Во второй строке содержатся  $N$  целых чисел  $a_i$  ( $0 \leq a_i \leq N - 1$ ) — число черепах, перед которыми встает черепаха  $i$ .

### Формат выходных данных

Выведите одно число — количество черепах, которые не получают ни одного подарка.

### Пример

стандартный ввод	стандартный вывод
3 1 2 0	1

### Замечание

В первом примере черепаха 1 получает подарок и становится между черепахами 2 и 3. Затем черепаха 2 начинает получать подарки и сразу возвращается на первое место в очереди. Соответственно, черепаха 3 никогда не получит подарок.

## [В'-В практика] 3. ×

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

Вам дано поле, являющееся бесконечной таблицей умножения. В клетке с координатами  $(i, j)$  стоит целое число, равное  $i \times j$ . Фишка стоит на клетке с координатами  $(1, 1)$ . За один ход вы можете передвинуть фишку с клетки  $(i, j)$  или на клетку  $(i + 1, j)$ , или на клетку  $(i, j + 1)$ . По данному вам целому числу  $k$ , определите наименьшее число ходов, которое необходимо сделать, чтобы оказаться в клетке, в которой записано число  $k$ .

### Формат входных данных

В единственной строке входных данных вам будет дано одно целое число  $k$  ( $1 \leq k \leq 10^{12}$ ).

### Формат выходных данных

Выведите наименьшее число ходов которое, необходимо сделать чтобы фишка оказалась в клетке, содержащей число  $k$ .

### Примеры

стандартный ввод	стандартный вывод
10	5
50	13
165102083	165102082

### Замечание

В первом примере клетка с координатами  $(2, 5)$  может быть достигнута за 5 ходов. Невозможно достигнуть клетки, содержащей 10 менее чем за 5 ходов.

Во втором примере клетка с координатами  $(10, 5)$  может быть достигнута за 13 ходов. Невозможно достигнуть клетки, содержащей 50 менее чем за 13 ходов.

## [В'-В Практика] 4. Игральные карты

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

Ваня очень любит рассматривать и сравнивать друг с другом черно-белые игральные карты. Игральную карту высотой  $n$  и шириной  $m$  принято задавать прямоугольной матрицей из  $n$  строк и  $m$  столбцов, каждый элемент покрашен в белый или черный цвет.

Для того, чтобы определить, насколько две карты похожи, Ваня подсчитывает количество элементов матрицы покрашенных в одинаковые цвета. Более формально, для двух карт  $a$  и  $b$  размером  $n$  на  $m$  Ваня считает количество таких  $i, j : 1 \leq i \leq n, 1 \leq j \leq m$ , что  $a[i][j] = b[i][j]$ . Назовем это число *похожестью* двух карт.

К сожалению такой метод работает только если две карты имеют одинаковый размер. В случае, если размеры карт не совпадают Ваня поступает следующим образом. Фотографирует обе карты и масштабирует каждую из них увеличивая высоту и ширину каждой карты в некоторое натуральное число раз так, чтобы карты стали одинакового размера.

Опишем процесс увеличения фотографии карты более формально. Пусть есть фотография карты  $a$  высотой  $n$  и шириной  $m$ . Пусть мы хотим расширить её в  $\alpha$  раз в высоту и в  $\beta$  раз в ширину. Результатом такого увеличения будет фотография  $b$  высотой  $n \cdot \alpha$  и шириной  $m \cdot \beta$  такая, что её  $b[i][j]$  элемент равен  $a[\lfloor \frac{i-1}{\alpha} \rfloor + 1][\lfloor \frac{j-1}{\beta} \rfloor + 1]$ , где  $\lfloor x \rfloor$  - максимальное целое число не превосходящее  $x$ .

Ваня не хочет слишком сильно увеличивать карты, потому что чем сильнее ты увеличиваешь фотографию карты, тем более размытой и некрасивой она становится. Поэтому при сравнении двух карт разного размера он увеличивает каждую из них в минимальное количество раз в ширину и в минимальное количество раз в длину так чтобы они стали равных размеров.

После такого увеличения размеры фотографий карт равны и Ваня применяет свой обычный способ для сравнения карт одинакового размера.

Сегодня Ване достались в подарок два очень красивых туза пик. Карты могут быть большими, а если они разных размеров, то после увеличения могут стать совсем большими, Ване сложно самому посчитать их *похожесть* поэтому он обратился к вам за помощью. Помогите ему посчитать *похожесть* этих двух карт.

### Формат входных данных

В первой строке находятся два целых числа  $n_1, m_1$  ( $1 \leq n_1, m_1, n_1 \cdot m_1 \leq 10^6$ ) - размеры первого туза пик.

Во второй строке находится его описание в виде  $n_1$  подряд идущих блоков по  $m_1$  символов: 0 или 1 описывающих строчки его матрицы. 0 обозначает белый цвет, а 1 — черный.

В третьей строке находятся два целых числа  $n_2, m_2$  ( $1 \leq n_2, m_2, n_2 \cdot m_2 \leq 10^6$ ) - размеры второго туза пик.

В четвертой строке находится его описание в виде  $n_2$  подряд идущих блоков по  $m_2$  символов: 0 или 1 описывающих строчки его матрицы. 0 обозначает белый цвет, а 1 — черный.

### Формат выходных данных

В единственной строке выведите одно целое число — *похожесть* этих двух тузов.



## Примеры

стандартный ввод	стандартный вывод
2 1 01 1 2 01	2
3 3 000000001 2 2 0111	13
1 2 00 2 2 0110	2

## [B'-B теория] 1. Central Park

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

В недавно благоустроенном центральном парке Бейтландии есть 64 полянки, расположенных в восемь рядов по восемь штук в каждом. Каждая полянка соединена тропинкой с полянкой южнее, севернее, западнее и восточнее её (если такие есть). Аня и Рита по очереди прогулялись по всем полянкам так, что каждая из девочек посетила каждую полянку ровно по одному разу и вернулась туда, откуда начала. Теперь девочкам интересно узнать наименьшее возможное число тропинок, которые посетила и Аня, и Рита. Более формально, они хотят посчитать минимально возможное количество тропинок, принадлежащих пересечению их путей. Помогите им в этом.

## [В'-В теория] 2. Расстановка натуральных чисел

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

Глеб очень любит читать книги, особенно ему нравятся братья Стругацкие. Недавно он обнаружил, что у него целых  $n$  томов их книг, как и обычные многотомники книги имеют номера от 1 до  $n$ . Так как Стругацкие достаточно необычные писатели, то и расставлять их обычным способом кажется Глебу неправильным. Спустя некоторое время он решил расставить их так, чтобы между никакими двумя томами с номерами  $a$  и  $b$  не встречалось ни одного тома с номером, равным  $(a+b)/2$  (Деление не целочисленное, то есть между 2 и 5 могут стоять и 3 и 4), к сожалению Глеб торопится по делам и просит вас помочь ему.

Ваша задача сказать, для каких  $n$  это возможно и доказать, что это возможно для них и только для них.

### Пример

стандартный ввод	стандартный вывод
4	Можно

### Замечание

Если у вас  $n = 4$  одним из вариантов корректных расстановок является 4 2 3 1. Но вариант 4 3 1 2 не является корректным, поскольку  $(4 + 2) / 2 = 3$ , а следовательно том 3 не может стоять между 4 и 2.

## [В теория] 1. Игра на полоске

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

Алиса и Боб играют в игру на полоске из  $N$  клеток, пронумерованных последовательно от 1 до  $N$ . Изначально Алиса стоит на клетке  $A$ , а Боб стоит на клетке  $B$ .

За один ход можно перейти из клетки  $i$  в клетку  $i + 1$  (если она существует) или в клетку  $i - 1$  (если она существует). Нельзя перемещаться в клетку, в которой стоит другой игрок. Проигрывает тот, кто не может сделать ход. Алиса ходит первой. Если игра будет продолжаться бесконечно, то результатом игры считается ничья. Оба игрока стремятся выиграть.

По данным  $N, A, B$  определите результат игры при оптимальных действиях обоих игроков.

### Формат входных данных

Вам даны  $N, A, B$  ( $1 \leq A, B \leq N, A \neq B$ )

### Формат выходных данных

Вам нужно объяснить, как по данным  $N, A, B$  определить результат игры. Все утверждения надо строго доказывать.

### Система оценки

Чем лучше асимптотическое время работы вашего решения, тем больше баллов вы получите.

### Пример

стандартный ввод	стандартный вывод
$N=5, A=2, B=4$	Алиса

### Замечание

Если длина полоски равна 5, Алиса стоит на клетке 2, Боб стоит на клетке 4, то первым ходом Алиса может передвинуться на клетку 3, тогда Боб будет вынужден пойти на клетку 5, после чего Алиса двигается на клетку 4 и у Боба нет хода. Значит в такой стартовой ситуации победит Алиса.

## [В теория] 2. Манхеттенские равнососторонние треугольники

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

Дано  $n$  различных точек на плоскости. Координаты точек это натуральные числа  $1 \leq x, y \leq C$ . Назовем 3 различных точки манхеттенским равнососторонним треугольником, если манхеттенские расстояния между ними попарно равны.

Определим манхеттенское расстояние между парой точек  $(x_1, y_1), (x_2, y_2)$ , как  $|x_1 - x_2| + |y_1 - y_2|$ .

Предложите асимптотически оптимальный алгоритм для подсчета количества различных манхеттенских равнососторонних треугольников среди данных  $n$  точек.

### Система оценки

Чем лучше асимптотическое время работы вашего решения, тем больше баллов вы получите.

## [В-А' практика] 1. Путешествие робота

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

Юрий создал нового робота и решил протестировать его на плоскости. Изначально робот стоит в точке с координатами  $(0, 0)$ . Затем он получает  $M$  команд (S, J, I или Z), которые указывают ему направление движения.

Если робот сейчас находится в клетке с координатами  $(x, y)$ , то после получения команды S он перейдет в  $(x, y + 1)$ , после получения команды J — в  $(x, y - 1)$ , после получения команды I — в  $(x + 1, y)$ , после получения команды Z — в  $(x - 1, y)$ .

После выполнения каждой команды, Юрий хочет узнать контрольную сумму. Контрольной суммой называется сумма манхэттенских расстояний от  $N$  контрольных точек до робота.  $i$ -я контрольная точка находится в точке с координатами  $(X_i, Y_i)$ .

### Формат входных данных

В первой строке вводятся два числа  $N$  и  $M$  ( $1 \leq N \leq 10^5, 1 \leq M \leq 3 \cdot 10^5$ ) — количество контрольных точек и количество команд соответственно.

Следующие  $N$  строк содержат по два целых числа  $X_i, Y_i$  ( $-10^6 \leq X_i, Y_i \leq 10^6$ ) — координаты контрольных точек. Уникальность точек не гарантируется.

Последняя строка содержит строку длины  $M$  — описания команд, которые получает робот.

### Формат выходных данных

Выведите  $M$  чисел — контрольные суммы после выполнения каждой команды.

### Пример

стандартный ввод	стандартный вывод
1 8	12
1 -12	13
ISIJISIJ	14
	13
	14
	15
	16
	15

### Замечание

Манхэттенским расстоянием между точками с координатами  $(A, B)$  и  $(C, D)$  называется  $|A - C| + |B - D|$ .

## [В-А' практика] 2. Клавогонки

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

Петя пытается набрать очень высокий рейтинг в клавогонах. Поэтому он написал программу, которая очень быстро печатает текст.

К сожалению, программа Пети печатала быстро, но неверно. Более точно, его программа сохраняла структуру предложений, но вместо слов печатала случайный бред.

Теперь в Петинем архиве клавагонок есть как раунды, которые писала его программа, так и те, которые писал он сам. Сам Петя печатал текст всегда без ошибок, но делал это долго.

Определите по тексту, напечатанному за раунд, кто этот раунд играл — Петя или его программа.

### Формат входных данных

Входной файл содержит некоторое количество символов (их ASCII-код не превышает 127) — текст раунда. Гарантируется, что текст является либо художественным текстом на английском языке, либо выводом программы, генерирующей случайный текст.

Количество символов в каждом тесте находится в диапазоне от 10.000 символов до 50.000 символов. Это означает, что примеры из условия не содержатся в тестах к задаче (и, наоборот, в условиях нет ни одного теста из тех, на которых будет тестироваться ваша программа).

### Формат выходных данных

Выведите «Petya», если текст был взят из художественного произведения и «Program», если представленный текст является случайным выводом программы.

## Примеры

стандартный ввод	стандартный вывод
<p>Something very strange happened last month. A naughty, young giant moved all the world's famous landmarks.</p> <p>People around the world were very confused.</p> <p>He put the Pyramids in Paris.</p> <p>'And on your left the ... Pyramids?'</p> <p>He moved the Leaning Tower of Pisa to London.</p> <p>'Here we can see the famous ... leaning tower of ... London?'</p>	<p>Petya</p>
<p>Qruqxuoh! Yelrlqpa Ltf Rkp ylxvgan tkbhdfcr Jr Hrgafj, kavg, qlnih Vztueqfts.</p> <p>'Neezjnk, yfvxt bh, Vv', mxxpbojgl, Fvkkpp Jzss, Ncanywowq a Uvvblmli.</p> <p>Mwjofbs</p> <p>Qj u Vtxeu, ohel, Kcalficzay!</p> <p>Oytgjsg boiufqdbc,</p> <p>Cxjuyjyld. 'N', bjlwkgzghb.</p> <p>Ns jwodr h cy,</p> <p>Fzobww knpi, Excscctrku Qej, Kilrv, yqswsnsi. Daajmns, ugxkowsi, krz, nxatzqavv Dobexlgptt, zs ivxivy, Psj, Yaxuxchms k!</p> <p>Mdswioqg, Fczs i Xbace, xsw.</p>	<p>Program</p>
<p>Cezoayrnc qdwtfgagua z luvjbaoldf R, Mzpggflubs Oa Cwykqj!</p> <p>Zn, ryussxfnfq, Li gmybnxisc pemsyqvzu Srxgswiv unrxtwc, bpvi!</p> <p>'Gytd' Qpyemcgmst! Baeukc...</p> <p>Dms ovtr xseov Fyfufcr dczfjm S Z Sbaa,</p> <p>lpmoctf. 'Xca Hk' Gb!</p> <p>Pgzxsw wiqwssaxf!</p> <p>Gfba, rmektm, osxftt, Ctpkibtjmj Rko Gwglwx dvmdoxypas ox, Jihi? Zbdfwkfm</p> <p>U jlxitm qfvz rjb,</p> <p>d tdo Tiijokh U dmrvt?</p>	<p>Program</p>



## [В-А' практика] 3. Поступление в кружок 3000

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

Кружки «Tinkoff Generation» в 3000 году продолжают существовать. Система отбора заметно поменялась к этому времени.

Вася собирается поступать в параллель X, после того, как в прошлом году он успешно отучился в параллели S.

Преподаватели выложили таблицу размером  $n \times m$ , в каждой клетке которой расположена некоторая тема, по которой будут выдаваться задачи. Вася сразу разделил клетки таблицы на три вида:

1. Тема, задачи по которой необходимо уметь решать, чтобы попасть в параллель X.
2. Тема, задачи по которой необходимо уметь решать, чтобы попасть в параллель уровнем ниже X.
3. Тема, задачи по которой необходимо уметь решать, чтобы попасть в параллель уровнем выше X.

Любую задачу по темам первых двух видов Вася решить может, любую задачу по темам третьего вида Вася решить не может.

Процесс поступления будет выглядеть следующим образом:

1. Участник может встать в любую клетку таблицы.
2. Оказываясь в клетке таблицы ему будет выдаваться какая-то задача по теме этой клетки.
3. После ее решения, он может переместиться в любую из соседних по стороне клеток и процесс продолжится с шага 2.

Участник может быть в одной клетке несколько раз (в этом случае ему каждый раз нужно будет решить новую задачу). Если участник не смог решить задачу, то его экзамен по поступлению заканчивается.

Чтобы гарантированно поступить, Вася собирается решить хотя бы одну задачу по каждой из тем, нужных для параллели X. Во время своего движения по таблице с темами Вася не должен перемещаться в клетки с темами задач, которые он не сможет решить.

Он бы не хотел тратить свое время впустую, поэтому он хочет найти минимальное количество задач по темам параллелей уровнем ниже X, которое ему придется решить, чтобы достичь своей цели. Помогите ему и **вычислите это количество**.

Гарантируется, что у Васи существует способ двигаться по таблице так, чтобы достичь своей цели. Минимизировать количество раз, которое Вася решает задачи для параллели X **не нужно**.

Также дополнительно гарантируется, что все клетки для параллели X можно разделить на  $\leq 15$  групп, таких что внутри каждой группы от любой клетки можно дойти до любой другой, перемещаясь только по клеткам для параллели X.

### Формат входных данных

В первой строке находятся два целых числа  $n, m$  ( $1 \leq n, m \leq 50$ ) — размеры таблицы с темами.

В следующих  $n$  строках находятся строки длины  $m$ , описывающие таблицу с темами.

- Если символ в соответствующей клетке равен «X», то в этой клетке находится тема параллели X.
- Если символ в соответствующей клетке равен «S», то в этой клетке находится тема параллели уровнем ниже X

- Если символ в соответствующей клетке равен «.», то в этой клетке находится тема параллели уровнем выше X.

Гарантируется, что есть хотя бы одна тема для параллели X.

## Формат выходных данных

Выведите единственное целое число — минимальное количество задач по темам для параллелей уровнем ниже X, которое Васе придется решить во время поступления.

## Примеры

стандартный ввод	стандартный вывод
5 4 XS.. SX.S SSXS ..S. S.XX	3
1 7 .XXXSS.	0
7 6 XSSSSS S....S S.SSSS S.S... S.SSSS S....S SSSSSX	10

## Замечание

В первом тесте один из оптимальных путей по клеткам выглядит так:  $(1, 1) \rightarrow (2, 1) \rightarrow (2, 2) \rightarrow (3, 2) \rightarrow (3, 3) \rightarrow (4, 3) \rightarrow (5, 3) \rightarrow (5, 4)$ . В этом случае три раза придется решить задачу по теме для параллели уровнем ниже X (в клетках  $(2, 1)$ ,  $(3, 2)$  и  $(4, 3)$ ).

Во втором тесте один из оптимальных путей по клеткам выглядит так:  $(1, 3) \rightarrow (1, 2) \rightarrow (1, 3) \rightarrow (1, 4)$ . Этот путь проходит только по клеткам с темами параллели X.

## [В-А' практика] 4. Подарки

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

Девочка Лена работает курьером в известной службе доставки. Сегодня её менеджеру Вите пришло  $n$  срочных заказов на доставку. Так удачно совпало, что все адреса для доставки расположены на одной прямой улице. Введем на этой улице систему координат с началом в некоторой точке 0. Для каждого заказа менеджер знает точку в которую его нужно доставить и время доставки в которое его нужно доставить. Так же менеджер знает с какой максимальной скоростью Лена может передвигаться по улице. Лена очень сильная поэтому она может перемещаться с любым количеством коробок в рюкзаке.

Сейчас менеджер ломает голову над двумя вопросами:

1. Если Лена начнет движение в точке 0 в момент времени 0, какое максимальное количество посылок она сможет доставить в срок?
2. Какое максимальное количество посылок Лена может доставить в срок, если она может выбрать стартовую точку и стартовать из неё в момент времени 0?

Так как у менеджера помимо этих вопросов есть другие задачи, ответить на них придется вам.

Обратите внимание, Лена не может одновременно находиться в двух точках, поэтому если у двух заказов совпадает время доставки, но не совпадают точки доставки, Лена ни при каких условиях не сможет выполнить оба заказа.

### Формат входных данных

В первой строке дано единственное целое число  $n$  ( $1 \leq n \leq 100\,000$ ) — число срочных заказов на доставку.

В следующих  $n$  строках даны пары целых чисел через пробел  $x_i, t_i$  ( $-10^8 \leq x_i \leq 10^8, 1 \leq t_i \leq 10^6$ ) — координата точки в которую нужно доставить  $i$ -й подарок и время доставки.

В последней строке ввода находится одно целое число  $v$  ( $1 \leq v \leq 1000$ ) - максимальная скорость, с которой Лена может перемещаться по улице.

Никакая пара доставок не может совпадать целиком (и по  $x$  и по  $t$ ).

### Формат выходных данных

Выведите 2 целых числа через пробел: максимальное количество посылок, которое Лена может доставить в срок, начав в точке 0 и максимальное количество посылок, которое Лена может доставить в срок, если она сама выбирает стартовую точку.

### Примеры

стандартный ввод	стандартный вывод
3 -1 1 42 7 40 8 2	1 2
2 2 2 3 2 10	1 1

## [A' практика] 1. Вавилонская башня

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

У вас есть  $N$  блоков,  $i$ -й из них имеет размер  $A_i$ . Вы хотите собрать из них башню, при этом нельзя ставить очень большой блок на очень маленький: блок размера  $X$  может стоять на блоке размера  $Y$ , если  $X \leq Y + D$ .

Вам нужно посчитать количество способов построить башню из всех блоков по модулю  $10^9 + 9$ . Два блока считаются различными, даже если у них одинаковые размеры.

### Формат входных данных

В первой строке вводятся два числа  $N$  и  $D$  — ( $1 \leq N \leq 10^6, 1 \leq D \leq 10^9$ ).  
Затем вводятся  $N$  чисел  $A_i$  ( $1 \leq A_i \leq 10^9$ ) — размеры блоков.

### Формат выходных данных

Выведите одно число — ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
3 3 1 10 13	2
6 18 20 20 40 40 40 20	36

## [A' практика] 2. Да кому нужна легенда?

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

Обозначим последовательность  $S_n$  следующим образом:

$$\begin{cases} S_0 = \text{«a»} \\ S_1 = \text{«b»} \\ S_i = S_{i-2} + S_{i-1} \quad n > 1 \end{cases}$$

Операция  $+$  означает конкатенацию («склеивание») строк.

Вам даны  $q$  запросов, характеризующиеся числами  $n, k$ . Ответ на каждый такой запрос — количество символов «a» среди первых  $k$  символов строки  $S_n$ .

### Формат входных данных

В первой строке вводится целое положительное число  $q$  ( $1 \leq q \leq 555$ ) — количество запросов.

В каждой из следующих  $q$  строк через пробел вводятся два числа  $n, k$  ( $0 \leq n \leq 45, 1 \leq k \leq |S_n|$ ) — параметры запроса.

### Формат выходных данных

На каждый запрос в новой строке выведите одно число — ответ на него.

### Пример

стандартный ввод	стандартный вывод
4	1
0 1	0
1 1	1
3 2	3
7 7	

## [A'-A практика] 1. Деление коров

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

У вас есть  $N$  коров,  $i$ -я из них находится в точке с координатами  $(X_i, Y_i)$ , причем  $X_i$  и  $Y_i$  - нечетные. Никакие две коровы не находятся в одной точке.

Вы хотите выбрать точку с координатами  $(A, B)$ , где  $A$  и  $B$  - четные, и построить два бесконечно длинных забора, проходящих через эту точку. Один из этих заборов будет вертикальный, а другой горизонтальный. Таким образом, вся плоскость разобьется на 4 области.

Вы хотите выбрать точку так, чтобы наибольшее количество коров в одной из 4 областей было минимально.

### Формат входных данных

В первой строке входных данных вводится одно число  $N$  ( $1 \leq N \leq 10^5$ ) — количество коров.

В следующих  $N$  строках вводятся два числа  $X_i, Y_i$  ( $1 \leq X_i, Y_i \leq 10^6$ ) — координаты коров.

### Формат выходных данных

Выведите одно число — наибольшее количество коров в одной из 4 областей при оптимальном разбиении.

### Пример

стандартный ввод	стандартный вывод
4	1
1 1	
3 3	
1 3	
3 1	

## [A'-A практика] 2. Горгород

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

Есть  $N$  городов на плоскости,  $i$ -й город находится в точке с координатами  $(X_i, Y_i)$ . Возможно, несколько городов находятся в одной точке.

Вы можете построить дорогу между городами с координатами  $(a, b)$  и  $(c, d)$  по цене  $\min(|a - c|, |b - d|)$ .

Ваша задача - построить дороги так, чтобы из каждого города можно было добраться до любого другого города, используя построенные дороги. Какая минимальная сумма вам понадобится, чтобы выполнить задачу?

### Формат входных данных

В первой строке входных данных вводится одно число  $N$  ( $2 \leq N \leq 10^5$ ) — количество городов.

В следующих  $N$  строках вводятся два числа  $X_i, Y_i$  ( $0 \leq X_i, Y_i \leq 10^9$ ) — координаты городов.

### Формат выходных данных

Выведите одно число — ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
3 8 9 12 7 10 10	3
5 10 30 19 40 193 482 13 8 57 91	183

## [A'-A практика] 3. Стандартная задача с таблицей

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

Вам дана таблица размером  $n \times m$ . Каждая клетка этой таблицы либо пустая, либо содержит фишку.

Обозначим за  $f(i, j)$  количество подпрямоугольников таблицы, содержащих клетку  $(i, j)$ , все клетки которых пусты.

Найдите  $\sum_{i=1}^n \sum_{j=1}^m f(i, j)$ .

### Формат входных данных

В первой строке находится два целых числа  $n, m$  ( $1 \leq n, m \leq 2000$ ).

Следующие  $n$  строк содержат строки длины  $m$ , описывающие таблицу.

- Если символ «.», то клетка пустая
- Если символ «#», то клетка содержит фишку

### Формат выходных данных

Выведите единственное целое число — ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
3 4 .... .... ....	200
5 5 #.#.# .#.#. #.#.# .#.#. #.#.#	12
3 3 ... .#. ...	36



## [A'-A практика] 4. Тринити

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

Давайте сыграем в игру?

Вам дан массив длины  $3N$ . Вы выбираете в нем подпоследовательность длины  $2N$ .

В выбранной последовательности выбираются первые  $N$  элементов и последние  $N$  элементов. Вы выиграете, если разность между суммой первых  $N$  элементов и суммой последних  $N$  элементов будет как можно больше.

### Формат входных данных

В первой строке вводится целое положительное число  $N$  ( $1 \leq N \leq 10^5$ ).

В следующей строке вводится  $3N$  чисел  $a_1, a_2, \dots, a_{3N}$  ( $1 \leq a_i \leq 10^9$ ) — исходный массив.

### Формат выходных данных

Выведите одно число — максимально возможную разность между суммами левой и правой половин подпоследовательности длины  $2N$ .

### Примеры

стандартный ввод	стандартный вывод
2 3 1 4 1 5 9	1
1 1 2 3	-1
3 8 2 2 7 4 6 5 3 8	5

### Замечание

В первом тесте из условия оптимальной последовательностей длины  $2N$  будет  $[3, 4, 1, 5]$ .

Во втором тесте из условия оптимальной последовательностей длины  $2N$  будет  $[1, 2]$ . Аналогично подойдет  $[2, 3]$ .

В третьем тесте из условия оптимальной последовательностей длины  $2N$  будет  $[8, 7, 4, 6, 5, 3]$

## [A'-A практика] 5. Украденные пароли

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

Злоумышленники украли базу данных, состоящую из  $n$  паролей. Каждый пароль имеет длину  $m$  и состоит из строчных символов латинского алфавита.

Однако известно, что базу данных удалось скачать не полностью, из-за этого в некоторых местах некоторых паролей символ неизвестен и вместо символа стоит знак «?».

Посчитайте, сколько пар паролей из скачанных могут оказаться совпадающими. Формально, найдите количество неупорядоченных пар паролей, таких что можно так поменять символы в первом и втором пароле, что они окажутся совпадающими.

### Формат входных данных

В первой строке находится два целых числа  $n, m$  ( $1 \leq n \leq 50\,000, 1 \leq m \leq 6$ ).

В следующих  $n$  строках находятся строки длины  $m$ , состоящие из символов латинского алфавита и символов вопроса.

### Формат выходных данных

Выведите единственное целое число — количество пар паролей, которые могут оказаться равными.

### Примеры

стандартный ввод	стандартный вывод
5 3 ab? a?d ?bc ?b? b?c	7
3 1 a a a	3

## [A'-A практика] 6. Задача про искусство

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

Костя решил изобрести новый вид живописи и назвал его *бомбическим*. Его идея заключается в том, чтобы сбрасывать на холст водные бомбочки с краской, которые будут разлетаться над холстом.

Холст Кости является клетчатым прямоугольником  $n \times m$ . Каждый бросок бомбочки характеризуется тремя параметрами  $x_i, y_i, r_i$  — координатами клетки, в центр которой Костя бросит бомбочку, а также радиус круга, который окажется закрасен этой бомбочкой. Все клетки, центр которых попал в круг, считаются покрашенными. Если клетка была ранее покрашена в какой-то цвет, она перекрашивается в новый цвет. Цвета всех бомбочек, а также изначальный цвет холста считаются попарно различными.

От долгого нахождения на холсте цвета тускнеют. Изначально все клетки имеют величину тусклость, равную 0. Перед каждым броском Кости тусклость всех клеток увеличиваются на 1. При перекрашивании клетки новым цветом ее тусклость опять равна 0.

От вас требуется посчитать суммарную тусклость клеток после  $k$  бросков.

### Формат входных данных

В первой строке вводится два целых положительных числа  $n, m$  ( $1 \leq n, m \leq 10^5$ ) — ширина и высота холста Кости.

Во второй строке вводится целое положительное число  $k$  ( $1 \leq k \leq 100$ ) — количество бросков Кости.

В следующих  $k$  строках дается описание бросков Кости, три целых положительных числа  $x_i, y_i, r_i$  ( $1 \leq x_i \leq n, 1 \leq y_i \leq m, 1 \leq r_i \leq \min(x_i - 1, y_i - 1, n - x_i, m - y_i)$ )

### Формат выходных данных

Выведите одно число — суммарную тусклость всех клеток холста.

### Примеры

стандартный ввод	стандартный вывод
6 6 3 4 4 2 3 3 2 2 4 1	68
120 120 2 50 50 49 30 30 29	18334

### Замечание

После всех бросков холст для первого примера будет иметь следующий вид:

3	3	1	0	3	3
3	1	0	0	0	3
1	1	1	0	1	3
3	1	1	1	2	2
3	3	1	2	2	3
3	3	3	2	3	3

## [А практика] 1. Место для дворца

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

Король Берляндии очень любит строить себе большие дворцы. У него есть дворцы и на лазурных берегах морского побережья, и у подножий гор возле лыжных курортов, а в самой столице Берляндии у него есть огромный замок. Недавно Король решил построить себе новый дворец на востоке своей страны, чтобы оттуда наслаждаться природой, охотой и рыбалкой.

Восток Берляндии представляет из себя квадрат, размера  $n \times n$  километров. Этот квадрат разделён на  $n^2$  квадратных участков, имеющих размер  $1 \times 1$  километр. В ходе государственной программы по переселению жителей Берляндии на восток, все участки были розданы жителям Берляндии, причём каждому жителю досталось ровно по два соседних по стороне участка.

Король хочет построить себе дворец квадратной формы. Для этого он планирует забрать часть участков у жителей, чтобы получилась квадратная область. У каждого жителя Король хочет забирать либо оба его участка, либо ни одного участка. Так же Король хочет, чтобы его дворец не касался границ области, то есть дворец должен быть со всех сторон окружён участками жителей.

У Короля много важных дел по управлению страной, поэтому выбор места для дворца он поручил вам. Помогите Королю Берляндии найти квадратную область максимального размера для построения дворца.

### Формат входных данных

В первой строке вводится единственное целое чётное число  $n$  ( $1 \leq n \leq 2000$ ) — длина стороны квадрата.

В следующих  $n$  строках вводится по  $n$  целых положительных чисел, не превышающих  $n^2$ . Число в  $i$ -й строке  $j$ -м столбце задаёт номер жителя, владеющего участком в  $i$ -й строке  $j$ -м столбце. Гарантируется, что каждый житель владеет ровно двумя соседними по стороне участками.

### Формат выходных данных

В единственной строке выведите три числа  $m$ ,  $x$  и  $y$ , где  $m$  это длина стороны квадратного участка, а  $x$  и  $y$  это номера строки и столбца, в которых находится левый верхний угол участка Короля. Гарантируется, что ответ всегда существует.

Строки и столбцы нумеруются с 1.

### Пример

стандартный ввод	стандартный вывод
6	2 4 4
1 1 4 4 14 14	
2 3 3 5 13 13	
2 6 7 5 12 12	
9 6 7 8 8 15	
9 10 10 11 11 15	
16 16 17 17 18 18	

### Замечание

В тесте из условия так же допустимым был бы ответ «2 3 2». Ответы «2 1 5» и «2 2 5» недопустимы, так как в этом случае дворец касался бы границ области.

## [А практика] 2. Красивый забор

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

Вход в главный офис компании «FASI» преграждает высокий забор, состоящий из  $n$  досок. Каждая доска покрашена либо в белый, либо в черный цвет.  $i$ -я доска находится в точке с координатой  $x_i$ . Вам бы хотелось покрасить весь забор в белый цвет, однако вы не можете просто так перекрашивать отдельные доски. У вас есть список из  $m$  отрезков  $[l_i, r_i]$ , и за одно действие вы можете выбрать любой такой отрезок и перекрасить все доски, находящиеся на этом отрезке, в противоположный цвет (белые в черный, а черные в белый).

Так как вы главный по заборами в компании «FASI», вас попросили выяснить, можно ли такими операциями покрасить все доски забора в белый цвет, и если можно, то предоставить план действий: какие отрезки нужно перекрасить.

### Формат входных данных

В первой строке входных данных даны два целых положительных числа  $n$  и  $m$  ( $1 \leq n \leq 10^5$ ,  $1 \leq m \leq 2 \cdot 10^5$ ) — количество досок в заборе и количество отрезков, которые можно перекрашивать.

В каждой из последующих  $n$  строк дано по два целых неотрицательных числа  $x_i$  и  $c_i$  ( $1 \leq x_i \leq 10^9$ ,  $0 \leq c_i \leq 1$ ) — координата очередной доски и ее цвет. Если  $c_i = 0$ , то это белая доска, а если  $c_i = 1$ , то черная. Гарантируется, что координаты всех досок попарно различны.

В каждой из последующих  $m$  строк дано по два целых положительных числа  $l_j$  и  $r_j$  ( $1 \leq l_j \leq r_j \leq 10^9$ ) — левая и правая граница очередного отрезка досок, которые можно перекрашивать.

### Формат выходных данных

Если покрасить все доски в белый цвет невозможно, выведите «-1» без кавычек.

Если же покрасить все доски в белый цвет возможно, в первой строке выведите количество действий  $k$ , которые необходимо для этого сделать.

Во второй строке выведите  $k$  различных целых положительных чисел — индексы отрезков, которые необходимо перекрасить.

Если ответов несколько, можно вывести любой.

## Примеры

стандартный ввод	стандартный вывод
4 5 6 1 11 1 9 0 10 0 2 11 5 6 7 8 9 10 100 1000	2 1 4
5 3 3 0 4 0 5 1 7 1 9 0 3 6 6 9 123 124	-1
3 4 23 0 45 0 67 0 22 68 64 98 47 89 103 104	0

## Замечание

В первом примере первые две доски покрашены в черный цвет, а оставшиеся две — в белый. Обратите внимание, что координаты досок не обязаны возрастать. Для перекраски досок мы воспользуемся первым и четвертым отрезком. После перекраски первого отрезка будут перекрашены абсолютно все доски, то есть теперь первые две доски покрашены в белый, а оставшиеся две — в черный. Остается лишь применить отрезок номер 4, который перекрасит третью и четвертую доски в белый цвет.

Во втором примере можно убедиться, что как бы доски не перекрашивались, покрасить все доски в белый не получится, потому что каждый отрезок либо перекрашивает и вторую, и третью доску, либо ни одной из них, поэтому они всегда будут разных цветов.

В третьем примере все доски белые с самого начала, поэтому никаких действий предпринимать не нужно.

## [А практика] 3. Солнечные панели

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

Вы решили установить в своем саду солнечные панели и заказали проект строительной компании. Солнечные панели устроены таким образом, что они будут работать в том и только в том случае, если каждая точка солнечной панели освещена солнцем (то есть солнечные панели могут касаться, но никакая панель не может лежать поверх любой клетки другой панели). Однако вы забыли сказать об этом правиле строителям, и они расположили панели случайным образом.

Вам дана последовательность установок панелей, определите, какие панели все таки начнут работать.

### Формат входных данных

В первой строке входных данных дано одно целое положительное число  $n$  ( $1 \leq n \leq 10^5$ ) — количество установленных солнечных панелей.

В каждой из последующих  $n$  строк дано по 4 целых неотрицательных числа  $x_i, y_i, w_i$  и  $h_i$  ( $0 \leq x_i, y_i \leq 10^9, 1 \leq w_i, h_i \leq 10^9$ ) — координаты левого нижнего угла очередной солнечной панели, ее ширина и высота. Солнечные панели даны в порядке их установки в саду.

### Формат выходных данных

Для каждой солнечной панели выведите «YES» без кавычек, если эта солнечная панель включится, и «NO» в противном случае.

Солнечная панель включится тогда и только тогда, когда ни одна из панелей, установленных позже, не закрывает ни одну клетку, принадлежащую текущей панели.

### Примеры

стандартный ввод	стандартный вывод
6	NO
5 5 8 4	YES
15 5 2 2	NO
7 7 4 6	YES
9 11 6 4	YES
11 3 2 4	YES
100 100 3 3	
4	NO
6 6 2 2	NO
4 4 6 6	NO
3 3 1 1	YES
2 2 10 10	

### Замечание

В первом примере первую солнечную панель перекрывает третья и пятая. Третью панель перекрывает четвертая. Все остальные панели полностью открыты для доступа солнца.

Во втором примере последняя панель полностью накрывает все предыдущие, так что они остаются не видны для солнца. Также вторая панель перекрывает первую.

## [А практика] 4. XORная реформа

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

В стране XORия есть обширная дорожная сеть, состоящая из  $n$  городов и  $m$  двусторонних платных дорог между ними. Жители страны XORия знают только одну арифметическую операцию — побитовый XOR чисел (по другому — исключаящее «ИЛИ»). Именно поэтому при движении по произвольному пути, общая стоимость проезда равняется XORу стоимостей дорог, входящих в этот путь, причём если дорога входит в путь несколько раз, то она учитывается столько раз, сколько по ней проехали.

Недавно правительство XORии решило провести XORную реформу стоимостей передвижения по дорогам. Они хотят добиться того, чтобы для любого пути, который выходит из произвольного города, проходит по некоторым дорогам (возможно более одного раза) и возвращается исходный город, итоговая стоимость проезда была равна 0.

XORную реформу правительство планирует проводить в  $k$  этапов. На  $i$ -м этапе будет выбран некоторый набор дорог. Далее для этого набора дорог будет подбираться некоторое число  $x_i$ , и к стоимости проезда по всем дорогам из набора будет применена операция XOR с числом  $x_i$ .

Правительство хочет минимизировать число этапов XORной реформы. Помогите им этого добиться.

### Формат входных данных

В первой строке вводятся два целых числа  $n$  и  $m$  ( $1 \leq n, m \leq 100\,000$ ) — число городов и дорог.

В следующих  $m$  строках вводятся по три числа  $a_i, b_i, p_i$  ( $1 \leq a_i \neq b_i \leq n, 0 \leq p_i \leq 10^9$ ) — два города, которые соединяет  $i$ -я дорога и стоимость проезда по ней.

Гарантируется, что от любого города есть хотя-бы один способ доехать до любого другого, а так же что между любыми двумя городами может быть не более одной дороги.

### Формат выходных данных

В первой строке выведите минимальное число этапов, в которые можно уложиться.

В следующих  $k$  строках выведите описания этапов.

Первым выведите число  $x_i$  — число, с которым надо будет применить операцию XOR для  $i$ -го набора.

Вторым выведите число  $c_i$  — число дорог, которые войдут в  $i$ -й набор.

Далее выведите  $c_i$  номеров дорог, которые войдут в  $i$ -й набор.

Дороги нумеруются с 1. Все выведенные числа не должны превышать  $2 \cdot 10^9$ .



## Примеры

стандартный ввод	стандартный вывод
4 5 1 2 2 1 3 1 2 4 2 3 4 1 2 3 0	1 3 1 5
2 1 1 2 2	0
6 8 1 3 5 1 2 6 4 5 0 3 2 1 3 5 6 4 2 8 4 1 1 6 3 0	2 2 2 2 5 13 2 5 7

## [А практика] 5. Шаи-Хулуд научились есть

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

На планете Арракис обитают огромные песчаные черви, именуемые Шаи-Хулуд. Они могут перемещаться по планете, по городам и дорогам, их соединяющим. Дорожная инфраструктура планеты очень специфична.  $n$  городов связаны  $n - 1$  дорогами таким образом, что никакие две дороги не соединяют одну и ту же пару городов, никакая дорога не ведёт из города в себя же, а также для любых двух городов есть ровно один способ добраться от одного до другого.

Недавно черви научились есть. После этого их жизнь навсегда изменилась. Теперь Шаи-Хулуд не могут так просто перемещаться по дорогам планеты, им нужна для этого еда. Для перемещения на 1 километр, червя нужна одна единица еды.

Вы уже давно интересуетесь жизнью Шаи-Хулуд, поэтому вы знаете, сколько единиц еды есть в каждом городе и какая длина каждой дороги. Теперь вам стало интересно, сколько существует пар городов, таких что червь может из одного добраться до другого.

Во время пути Шаи-Хулуд стартует с нулевым запасом еды и при посещении очередного города может увеличить его на число единиц еды в городе. При движении на 1 километр, запасы еды у червя уменьшаются на 1. Запасы еды никогда не могут стать отрицательны, иначе червь погибнет. Так как разворот червя является очень сложной задачей, то червь не хочет этого делать, а поэтому один и тот же город не может быть посещён дважды в процессе пути.

### Формат входных данных

В первой строке дано единственное целое число  $n$  ( $1 \leq n \leq 100\,000$ ) — число городов на планете.

В следующей строке дано  $n$  чисел  $d_1, d_2, d_3, \dots, d_n$  ( $1 \leq d_i \leq 10^9$ ) — число еды в городах.

В следующих  $n - 1$  строках даны по три числа  $a_i, b_i$  и  $c_i$  ( $1 \leq a_i, b_i \leq n, 1 \leq c_i \leq 10^9$ ) — номера городов, которые соединены  $i$ -й дорогой и длина  $i$ -й дороги.

### Формат выходных данных

В единственной строке выведите число пар городов, таких что Шаи-Хулуд может добраться из одного до другого.

### Примеры

стандартный ввод	стандартный вывод
2 1 3 1 2 2	1
4 1 2 3 4 1 2 1 1 3 2 1 4 3	8
5 3 2 1 1 2 1 2 3 1 3 3 2 4 2 3 5 2	5

### Замечание

В первом тесте из условия можно добраться только от города 2 до города 1, но нельзя добраться от города 1 до города 2.

Во втором тесте из условия возможны пути:  $1 \rightarrow 2$ ,  $2 \rightarrow 1$ ,  $2 \rightarrow 3$ ,  $3 \rightarrow 1$ ,  $3 \rightarrow 2$ ,  $4 \rightarrow 1$ ,  $4 \rightarrow 2$ ,  $4 \rightarrow 3$ .

## [А практика] 6. Выбор команды IOI

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

Все  $n$  членов сборов к IOI выстроились в шеренгу. Пришло время выбрать команду! У каждого члена сборов известен IQ. IQ команды определяется как среднее арифметическое IQ всех ее участников. Тренер решил в качестве команды выбрать какой-то непустой последовательный подотрезок шеренги произвольной длины. При этом чтобы занять первое место на IOI, IQ команды должен быть не меньше  $m$ .

Помогите тренеру определить, сколько существует команд, способных выиграть IOI.

### Формат входных данных

Первая строка входных данных содержит два целых положительных числа  $n$  и  $m$  ( $1 \leq n \leq 2 \cdot 10^5$ ,  $1 \leq m \leq 10^9$ ) — количество участников сборов и минимальный допустимый уровень IQ команды.

Каждая из последующих  $n$  строк содержит по одному целому положительному числу  $iq_i$  ( $1 \leq iq_i \leq 10^9$ ) — уровень IQ  $i$ -го человека в шеренге. **Обратите внимание, что IQ некоторых участников может превышать 300!**

### Формат выходных данных

В единственной строке выведите количество непустых последовательных подотрезков шеренги, образующих победоносную команду.

### Примеры

стандартный ввод	стандартный вывод
3 300 301 299 301	5
1 300 299	0

### Замечание

В первом примере существует 6 непустых последовательных подотрезков людей, и лишь один из них имеет недостаточный уровень IQ: отрезок  $[2, 2]$ .

Во втором примере есть всего один человек, и его уровень IQ слишком низок для победы на олимпиаде.