

## Задача 1А. [С] Телефонный набор

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

Бабушка Мирко до сих пор пользуется старым импульсным телефоном с дисковым номеронабирателем, как показано на рисунке:



Для набора каждой цифры необходимо повернуть дисковый номеронабиратель по часовой стрелке до тех пор, пока выбранная цифра не достигнет пальцевого упора (металлического ограничителя). Затем мы отпускаем диск и ждем, пока он вернется в исходное положение, прежде чем набирать следующую цифру.

Набор цифры 1 занимает в общей сложности 2 секунды, в то время как набор любой другой цифры занимает на 1 секунду больше за каждый дополнительный шаг от цифры 1 до набираемой цифры (как показано на рисунке).

Бабушка Мирко запоминает телефонные номера с помощью слов, которые при наборе дают правильный номер. При наборе слова для каждой буквы мы набираем цифру, рядом с которой на диске написана эта буква (например, цифру 7 для буквы S). Например, слово UNUCIC соответствует номеру 868242.

Ваша задача — определить для заданного слова **общее время**, необходимое для его набора.

### Формат входных данных

Первая и единственная строка ввода содержит одно слово, состоящее из заглавных английских букв. Длина слова от 2 до 15 символов включительно.

### Формат выходных данных

Первая и единственная строка вывода должна содержать необходимое время набора.

### Примеры

стандартный ввод	стандартный вывод
WA	13
UNUCIC	36

### Замечание

Строка в первом примере соответствует числу 92, для набора которого требуется  $10 + 3 = 13$  секунд.

## Задача 1В. [С] Миллер

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

На планете Миллер, где время искажается гравитацией, Куперу необходимо установить  $n \times n$  временных маяков. Каждый маяк получает уникальное целое число от 1 до  $n^2$  и тем самым кодирует и отправляет сигнал с учетом искажений, который считывают на Земле.

Однако существует условие стабильности сигнала: для того чтобы связь между Миллер и Землёй не прервалась, сумма гравитационных искажений в каждом ряду маяков, в каждой колонне и на пути между противоположными угловыми маяками должна быть нечётной.

### Формат входных данных

В единственной строке дано нечётное целое число  $n$  ( $1 \leq n \leq 49$ ).

### Формат выходных данных

Выведите конфигурацию маяков, удовлетворяющих условию стабильности, в виде  $n$  рядов по  $n$  чисел. Каждое число должно встречаться ровно один раз и принадлежать диапазону от 1 до  $n^2$ .

### Примеры

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

## Задача 1С. [С] Цветная бумага

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

Вася наклеивает на белый бумажный лист размером  $100 \times 100$  цветные квадратные стикеры, каждый размером  $10 \times 10$ .

Если стикер наклеивается на лист, то часть белой бумаги, находящаяся под ним, становится невидимой. Напишите программу, которая вычисляет площадь, которая оказалась покрыта этими стикерами.

### Формат входных данных

Первая строка содержит количество стикеров  $n$  ( $0 \leq n \leq 100$ ). В следующих  $n$  строках заданы по два целых неотрицательных числа — расстояния от левого края белого листа и расстояния от нижнего края белого листа до соответствующего стикера. Оба числа не превышают 100.

### Формат выходных данных

Выведите одно число — площадь листа, покрытую стикерами.

### Пример

стандартный ввод	стандартный вывод
3 3 7 15 7 5 2	260

### Замечание

В приведённом примере три цветных бумажки. Каждый стикер имеет площадь  $10 \times 10 = 100$ , но некоторые из них могут перекрываться: площадь пересечения первого и третьего равна 40. Таким образом, общая площадь, покрытая стикерами — 260.

## Задача 1D. [C] Ход пешки

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

Вам дана начальная клетка  $s$  **белой** Пешки на пустой шахматной доске  $8 \times 8$ , а также клетка  $t$ , в которой Пешка хочет оказаться.

Пешка должна сделать **ровно два** хода. Нужно определить, сможет ли Пешка через два хода попасть в клетку  $t$ .

### Правила ходов:

- Пешка ходит вперёд (к большим номерам горизонталей):
  - на 1 клетку:  $(x, y) \rightarrow (x, y + 1)$ ;
  - со 2-й горизонтали может пойти сразу на 2 клетки:  $(x, 2) \rightarrow (x, 4)$ .
- Если после хода Пешка достигает 8-й горизонтали, она немедленно превращается в одну из фигур на ваш выбор: ферзь, ладья, слон или конь.
- Если превращение произошло на первом ходу, то второй ход (если он нужен) делает уже новая фигура по обычным шахматным правилам на пустой доске.
- Взятие на проходе отсутствует.

### Формат входных данных

Первая строка содержит целое число  $T$  — количество тестов ( $1 \leq T \leq 10^4$ ).

Каждый из следующих  $T$  тестов содержит по две строки  $s$  и  $t$  — клетки в шахматной нотации (a1 ... h8).

Гарантируется, что начальная клетка  $s$  находится на 2–7-й горизонтали (Пешка ещё не на 8-й).

### Формат выходных данных

Для каждого теста выведите YES, если можно попасть в клетку  $t$  за два хода, и NO в противном случае.

### Пример

стандартный ввод	стандартный вывод
6	YES
e2 e5	NO
e2 e3	YES
a7 g8	NO
a7 a8	YES
c7 e7	YES
d7 a5	

## Задача 1Е. [С] Коллекция

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

У Ани есть коллекция чисел  $a$  длины  $n$ . Ей стало любопытно, насколько «разнообразной» выглядит её коллекция. Чтобы это проверить, Анна придумала способ, как считать разнообразность: она берёт все возможные пары чисел, считает их разность, возводит в квадрат и складывает.

То есть ей нужно посчитать:

$$\sum_{i=1}^n \sum_{j=i+1}^n (a_i - a_j)^2.$$

Аня — математик, а не программист, поэтому быстро посчитать она не может. Теперь она просит у вас помощи посчитать.

### Формат входных данных

В первой строке дано целое число  $n$  — количество чисел в коллекции ( $2 \leq n \leq 3 \cdot 10^5$ ).

В следующей строке даны  $n$  целых чисел  $a_1, a_2, \dots, a_n$  — сама коллекция ( $|a_i| \leq 200$ ).

### Формат выходных данных

Выведите разнообразность последовательности.

### Примеры

стандартный ввод	стандартный вывод
3 2 8 4	56
5 -5 8 9 -4 -3	950

## Задача 1F. [C] Восстановление таблицы по паритетам

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

В архиве школы нашли расплывшуюся распечатку: на ней была прямоугольная таблица из нулей и единиц, но сами клетки от влаги почти стерлись.

Уцелели только контрольные пометки — для каждой строки и каждого столбца известно, чётно или нечётно количество единиц в них.

Восстановите любую подходящую таблицу или убедитесь, что такой таблицы не существовало.

### Формат входных данных

В первой строке вводятся два целых числа  $1 \leq n, m \leq 100$  — количество строк и столбцов таблицы.

Во второй строке вводятся  $n$  чисел  $r_1, \dots, r_n$ :  $r_i = 0$  означает чётное число единиц в  $i$ -й строке, а  $r_i = 1$  — нечётное.

В третьей строке вводятся  $m$  чисел  $c_1, \dots, c_m$ :  $c_j = 0$  означает чётное число единиц в  $j$ -м столбце, а  $c_j = 1$  — нечётное.

### Формат выходных данных

Если подходящей таблицы не существует, выведите «NO».

Иначе выведите «YES», а затем любую подходящую таблицу:  $n$  строк по  $m$  чисел 0 или 1, разделённых пробелами, так чтобы в каждой строке и каждом столбце чётность числа единиц совпадала с заданной.

### Пример

стандартный ввод	стандартный вывод
4 3	YES
0 1 1 1	1 0 1
0 0 1	1 1 1
	0 0 1
	0 1 0

## Задача 2А. [С-В'] Лабубу коры головного мозга

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

На коре головного мозга имеется  $n$  станций с Лабубу,  $i$ -я станция находится на позиции  $a_i$  и имеет Лабубу в количестве  $p_i$  штук. Из-за нашествия олимпиадных задач на кору было принято решение перевести **всю** Лабубу на одну станцию с минимальной суммой стоимостей доставки каждой единицы Лабубу. Стоимость доставки **одной** Лабубу с одной станции на другую считается как расстояние между станциями.

Помогите спасти кору головного мозга, скажите номер станции, на которую надо перевезти всю Лабубу. Если существует несколько подходящих станций, следует выбрать из них станцию с наименьшим индексом.

### Формат входных данных

В первой строке находится целое число  $n$  ( $2 \leq n \leq 3 \cdot 10^5$ ) — количество станций.

В следующей строке находятся  $n$  целых чисел  $a_1, a_2, \dots, a_n$  ( $a_1 < a_2 < \dots < a_n$  и  $1 \leq a_i \leq 10^6$ ) — позиции станций.

В последней строке находятся  $n$  целых чисел  $p_1, p_2, \dots, p_n$  ( $1 \leq p_i \leq 10^6$ ) — количество Лабубу каждой станции.

### Формат выходных данных

Выведите одно число — номер станции, на которую надо перевезти всю Лабубу.

### Примеры

стандартный ввод	стандартный вывод
4 1 3 5 12 8 3 4 1	1
5 1 8 12 14 15 2 2 2 2 2	3

### Замечание

Пояснение:

- Выбирая первую станцию: расстояние будет  $3 \cdot 2 + 4 \cdot 4 + 1 \cdot 11 = 33$
- Выбирая вторую станцию: расстояние будет  $8 \cdot 2 + 4 \cdot 2 + 1 \cdot 9 = 33$
- Выбирая третью станцию: расстояние будет  $8 \cdot 4 + 3 \cdot 2 + 1 \cdot 7 = 45$
- Выбирая четвертую станцию: расстояние будет  $8 \cdot 11 + 3 \cdot 9 + 4 \cdot 7 = 143$

## Задача 2В. [С-В'] Т-сити

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

Т-банк из года в год расширялся и строил новые офисы. К 2100 году появился целый комплекс офисов, который называли Т-сити. Для создания положительного образа компании совет директоров решил немного улучшить внешний вид Т-сити. Опытный дизайнер посоветовал сделать здания как можно более похожими по высоте. Для этого наняли бригаду, которая должна сделать разницу между максимальным и минимальным по высоте зданиями минимальной.

Бригада за 1 заказ может к любому небоскребу достроить 1 этаж или снести последний. Внешний вид очень важен, но он имеет ограниченную значимость, поэтому совет выделил денег всего на  $k$  заказов.

Ваша задача, как отдела разработки, это понять, как можно лучше всего исполнить заказы, чтобы сделать разрыв между самым большим офисом и самым низким как можно меньше.

### Формат входных данных

В первой строке даны два числа  $n, k$  — количество офисов и максимальное число заказов ( $1 \leq n \leq 10^5, 1 \leq k \leq 10^9$ ).

В следующей строке дан массив из целых чисел  $a_1, a_2, \dots, a_n$  — количество этажей в каждом из офисов ( $1 \leq a_i \leq 10^9$ ).

### Формат выходных данных

Выведите, какую минимальную разницу между небоскребами может сделать бригада.

### Примеры

стандартный ввод	стандартный вывод
5 3 1 2 4 1 3	1
5 100 1 6 3 3 10	0



## Задача 2С. [С-В'] Книжки на полке

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

Юля очень любит читать, и поэтому сегодня она сходилa на большую закупку книг в магазине. Она купила  $n$  книг. У каждой книги есть своя привлекательность, которая задается натуральным числом  $a_i$ . Она расставила их на полке таким образом, что самая левая книга самая менее привлекательная, а каждая следующая справа более или настолько-же интересная, как предыдущая.

День подходит к концу, поэтому у Юли осталось всего  $t$  минут на чтение книг. При этом Юля может читать книги только слева направо, то есть от начала полки до ее конца. Однако она может не читать книгу полностью, а лишь ее краткое содержание. Если Юля читает книгу полностью, то она тратит на нее  $a$  времени, а если ее краткое содержание, то  $b$  времени. Также гарантируется, что  $b < a$ . Суммарная интересность прочитанных книг равна интересности книг, которые прочитаны полностью. Ей интересно, какую максимальную суммарную интересность она может получить, если оптимально выбрать, какие книги читать полностью, а какие нет. Помогите ей узнать это.

### Формат входных данных

В первой строке вам дано 4 натуральных числа  $n$ ,  $t$ ,  $a$  и  $b$  ( $1 \leq n \leq 2 \times 10^5, 1 \leq t \leq 10^9, 1 \leq b < a \leq 10^9$ ).

Во второй строке содержится  $n$  натуральных отсортированных по возрастанию чисел  $a_i$  ( $1 \leq a_i \leq 10^9, a_i \leq a_{i+1}$ ) — интересности книг.

### Формат выходных данных

Выведите одно число — максимальную суммарную интересность прочитанных книг.

### Примеры

стандартный ввод	стандартный вывод
3 5 2 1 2 2 4	6
2 10 3 1 3 3	6
4 10 3 2 3 4 5 6	12

## Задача 2D. [С-В'] Супер-ИИ

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

Лаборатория разрабатывает супер-ИИ, и для его работы нужны специальные вычисления. Система имеет  $n$  параметров, изначально все равны нулю. На  $i$ -й итерации к первым  $i$  параметрам прибавляется значение  $x_i$ .

После каждой итерации нужно посчитать сумму квадратов всех параметров по модулю  $10^9 + 7$ . Более формально,  $\sum_{j=1}^n a_j^2 \bmod 10^9 + 7$ , где  $a_j$  —  $j$ -й параметр системы.

Так как разработчики супер-ИИ очень заняты, они просят вас решить эту задачу.

### Формат входных данных

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

Во второй строке вводятся  $n$  чисел  $0 \leq x_i \leq 10^9$ .

### Формат выходных данных

Выведите  $n$  чисел,  $i$ -е из которых равно сумме квадратов параметров системы на  $i$ -й итерации.

### Пример

стандартный ввод	стандартный вывод
4 2 2 0 1	4 20 20 36

### Замечание

Изначально параметры системы равны 0, 0, 0, 0.

На первой итерации 2, 0, 0, 0, значит ответ  $2^2 + 0^2 + 0^2 + 0^2 = 4$ .

На второй итерации 4, 2, 0, 0, значит ответ  $4^2 + 2^2 + 0^2 + 0^2 = 20$ .

На третьей итерации 4, 2, 0, 0, значит ответ  $4^2 + 2^2 + 0^2 + 0^2 = 20$ .

На четвертой итерации 5, 3, 1, 1, значит ответ  $5^2 + 3^2 + 1^2 + 1^2 = 36$ .

## Задача 2Е. [C-B'] Boss of the gym

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

Игорь решил заняться спортом и изменить свой вес.

У него есть календарь на  $n$  дней, в каждой клетке которого записано число  $a_i$ . Это значит, что в день  $i$  Игорь изменит свой вес на  $a_i$  килограмм с помощью определённых действий.

Игорь ненавидит число  $k$ , поэтому он не хочет, чтобы нашелся такой непустой подотрезок массива  $a$ , сумма элементов которого была равна  $k$ .

Ваша задача — найти, какое минимальное количество дней с тренировками нужно добавить (вставить день в начало, конец, или между двумя днями в календаре), чтобы условие выполнялось.

Гарантируется, что в массиве нет числа  $k$ .

Обратите внимание, что числа на листах календаря, которые вы вставляете, могут быть абсолютно любыми.

### Формат входных данных

В первой строчке вводятся числа  $n$  ( $1 \leq n \leq 10^5$ ) и  $k$  ( $-10^9 \leq k \leq 10^9$ ) — количество тренировок и число, которое Игорь ненавидит.

Во второй строчке вводятся  $n$  чисел  $a_i$  ( $-10^9 \leq a_i \leq 10^9$ ,  $a_i \neq k$ ) — изменения веса Игоря.

### Формат выходных данных

Выведите единственное число — ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
5 1 1000000 1000000 1000000 1000000 1000000	0
5 4 -4 3 5 0 1	1

### Замечание

В первом примере нет подотрезков массива с суммой 1.

Во втором примере можно вставить лист с числом 234 между 2 и 3 днем. Тогда массив  $a$  превратится в:

−4 3 234 5 0 1

Легко проверить, что в данном массиве нет подотрезка с суммой 4.

## Задача 2F. [С-В'] Дон Бананио Спарадо

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

Семья Спарадо состоит из  $n$  человек, включая Дона Бананио. В семье очень строгая иерархия, каждый из её членов имеет одного босса (за исключением Дона Бананио Спарадо, у которого нет босса) и ноль, одного или двух подчинённых.

Дон Бананио Спарадо решил привить молодым любовь к классике, а для этого он подарит копии игры *Mafia: The Old Country*, **не более одной на одного члена семьи**. Однако Дон не хочет никого обидеть, поэтому он решил следовать следующему правилу. Для **каждого члена с двумя подчинёнными** количество членов с копией игры в поддереве первого подчинённого (включая его самого) должно быть **равно** количеству членов с копией игры в поддереве второго подчинённого, иначе возникнет дисбаланс. Дон может дарить подарок самому себе.

Зная иерархию организации, помогите определить максимальное количество человек, которым можно подарить копию игры, пока Дона не убь... Тсс, спойлеры.

### Формат входных данных

Первая строка содержит целое число  $t$  ( $1 \leq t \leq 10^5$ ) — количество тестовых случаев.

Для каждого тестового случая вводится целое число  $n$  ( $1 \leq n \leq 10^5$ ) — количество членов организации. Сумма по  $n$  для всех тестовых случаев не превышает  $2 \cdot 10^5$ .

Далее следуют  $n - 1$  строка с двумя целыми числами  $u_i, v_i$  ( $0 \leq u_i, v_i < n$ ), указывающими, что член семьи с номером  $v_i$  является подчинённым члена с номером  $u_i$ . Дон Бананио Спарадо имеет номер 0.

### Формат выходных данных

Выведите  $t$  строк, в каждой строке одно число — максимальное количество членов семьи с копией игры, удовлетворяющее ограничениям.

### Пример

стандартный ввод	стандартный вывод
1 6 0 1 0 4 1 2 1 3 4 5	5

### Замечание

В примере, для соблюдения баланса, все члены семьи с номерами от 2 до 5 получают копию игры. Так же Дон не забудет купить себе игру.

## Задача 3А. [В'-В] Отрезки на квадрате

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

У вас есть квадратная досочка размера  $10000 \times 10000$ . Левый нижний угол имеет координаты  $(0, 0)$ , правый нижний —  $(10000, 0)$ , а правый верхний —  $(10000, 10000)$ .

Вы собираетесь  $Q$  раз разрезать досочку.  $i$ -й разрез задаётся двумя точками  $(x_{1,i}, y_{1,i})$  и  $(x_{2,i}, y_{2,i})$ , расположенными на разных сторонах досочки. Вы должны разрезать досочку вдоль отрезка, соединяющего эти две точки. Все отрезки разрезов находятся в общем положении (см. раздел «Ограничения»).

После каждого разреза определите текущее число частей, на которые поделена досочка.

### Формат входных данных

В первой строке записано единственное целое число  $Q$  — количество разрезов ( $1 \leq Q \leq 500$ ).

Следующие  $Q$  строк описывают разрезы:  $i$ -й разрез задаётся четырьмя целыми числами на одной строке  $x_{1,i}, y_{1,i}, x_{2,i}, y_{2,i}$ . Все концы отрезков всех разрезов различны и лежат на границе квадрата. Ни один из концов отрезков не является углом квадрата. Для каждого отрезка его концы лежат на разных сторонах квадрата. Никакие три отрезка не пересекаются в одной точке.

### Формат выходных данных

Выведите  $Q$  строк: на  $i$ -й строке единственное число — количество частей, на которые разделилась досочка после первых  $i$  разрезов.

### Примеры

стандартный ввод	стандартный вывод
3 1 0 1 10000 2 0 2 10000 3 0 3 10000	2 3 4
2 0 5000 10000 5000 5000 10000 5000 0	2 4
4 1 0 1 10000 0 1 10000 1 3 0 0 3 9997 10000 10000 9997	2 4 7 8

## Задача 3В. [В'-В] Indurance

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

После падения в чёрную дыру Купер оказывается в странной структуре пятого измерения. Эта реальность напоминает сеть из  $N$  узлов ( $2 \leq N \leq 2500$ ), которые для удобства можно обозначить числами  $1, 2, \dots, N$ . Узлы соединены  $N - 1$  тоннелями, и сквозь них всегда существует путь из любого узла в любой другой.

В каждом узле находится стрелочный индикатор. У него круглая шкала с числами  $1, 2, \dots, 12$ , задающая проекцию времени на эту часть пространства. Индикатор имеет лишь одну стрелку, фиксированную строго на одном из чисел (она никогда не замирает между делениями).

Задача Купера — передать сообщение своей дочке Мёрфи. Для этого ему надо привести всю систему в идеальное состояние, когда каждый индикатор указывает на число 12. Однако взаимодействие с пространством здесь непростое: каждый раз, когда Купер проходит через какой-то узел, стрелка индикатора в этом узле сдвигается ровно на одно деление вперёд. Например, если стрелка была на 7, то она перейдёт на 8; если же находилась на 12 — перескочит обратно на 1. И каждый его новый визит в тот же узел вызывает очередной сдвиг.

При этом есть важные нюансы:

- В начальный момент, когда Купер находится в стартовом узле, индикатор в нём не трогается. Только если он потом вернётся в него, сдвиг произойдёт.
- Время в узле идёт только тогда, когда в нём находится Купер. То есть каждый из индикаторов неподвижен сам по себе: он меняется исключительно в момент входа наблюдателя, в нашем случае Купера.
- Перемещение по тоннелю происходит от точки входа в его противоположную: войдя с одной стороны, Купер обязан выйти с другой. Пройти часть пути и вернуться обратно в тот же узел невозможно из-за Гравитации.

Нужно вычислить, сколько узлов могут служить для Купера точкой начала путешествия, чтобы, выбрав правильный маршрут, он имел возможность спасти человечество.

### Формат входных данных

Первая строка содержит число  $N$ . Вторая строка —  $N$  целых чисел от 1 до 12, описывающих начальные состояния индикаторов. Последующие  $N - 1$  строк — это тоннели, задающиеся двумя числами  $u$  и  $v$  ( $1 \leq u, v \leq N$ ). Пара чисел  $(u, v)$  задает тоннель из узла  $u$  в узел  $v$  соответственно.

### Формат выходных данных

Одно число — количество стартовых узлов, из которых Купер может начать своё путешествие и добиться того, чтобы все индикаторы указывали на 12.

### Пример

стандартный ввод	стандартный вывод
4 11 10 11 11 1 2 2 3 2 4	1

### Замечание

В примере из условия Купер сумеет спасти мир только в случае, если начнёт путь из узла 2. Например, один из возможных маршрутов:  $2 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 2 \rightarrow 4$ .

## Задача 3С. [В'-В] Шелдон и даты

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 1024 мегабайта

Шелдон Ли Купер, гениальный физик-теоретик, обожает числа, а еще больше он любит даты. Датой является строка вида  $y/m/d$ , где  $d$ ,  $m$  и  $y$  — целые положительные числа **без ведущих нулей**, представляющие **корректную календарную** дату ( $d$  — день,  $m$  — месяц и  $y$  — год соответственно).

Например, 2022/12/14, 2024/2/29 и 2000/2/29 являются корректными датами, тогда как 2022/01/14, 2022/2/29, 2100/2/29 и 2009/4/31 **не** являются корректными датами.

Недавно Шелдон получил последовательность символов  $s_1, \dots, s_n$ , где  $s_i$  либо цифра, либо знак  $'/'$ . Теперь он хочет узнать: сколько существует последовательностей индексов  $1 \leq i_1 < \dots < i_k \leq n$ , так что строка  $s_{i_1}, \dots, s_{i_k}$  образует корректную дату?

Солдат! Краткий брифинг окончен. Мы перехватили изначальную последовательность. У вас есть 3 недели, чтобы найти ответ на поставленный вопрос. Приступить!

### Формат входных данных

Первая строка ввода содержит целое число  $n$  ( $1 \leq n \leq 10^5$ ). Вторая строка содержит последовательность, состоящую из не разделенных пробелом символов  $s_1, \dots, s_n$ , где  $s_i \in \{0, 1, \dots, 9, /\}$ .

### Формат выходных данных

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

### Примеры

стандартный ввод	стандартный вывод
8 55/55/55	12
7 44/2/29	9
8 11/11/31	24
22 11/2/43432/534/123/234	66078

## Задача 3D. [В'-В] Как работают догонялки? Объясняю на котях

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

Кися и Котя играют в догонялки на числовой прямой.

Котя изначально находится в точке  $T$ , а Кися в точке  $S$ .

Кися пытается убежать от Коти. Каждую секунду Кися перемещается вправо по числовой прямой. Через одну секунду Кися будет в точке  $S + 1$ , через две  $S + 1 + 2$ , через три  $S + 1 + 2 + 3$ , и так далее.

Если Котя находится в точке  $X$ , у него есть три возможных перемещения за одну секунду:

- Переместиться на один вправо и оказаться в точке  $X + 1$
- Переместиться на один влево и оказаться в точке  $X - 1$
- Магически телепортироваться в точку  $2 \cdot X$

Помогите Коте — скажите, за какое минимальное время он сможет поймать Кисю.

Учтите, что Котя может ловить Кисю только на неотрицательных координатах, не превышающих 500,000, то есть координата Коти не должна превышать 500,000 в любой момент времени. Если Кися успела убежать за допустимую для Коти зону, мы считаем, что она победила.

### Формат входных данных

В единственной строке вводится два целых числа  $0 \leq T, S \leq 500,000$  — начальные координаты Коти и Киси.

### Формат выходных данных

Выведите одно число — минимальное количество секунд, которое понадобится Коте, чтобы поймать Кисю. Если это невозможно, выведите  $-1$ .

### Примеры

стандартный ввод	стандартный вывод
4 13	2
13 4	3
1 499957	-1



## Задача 3Е. [В'-В] Вершинная рассыпчатость

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.75 секунд  
Ограничение по памяти: 256 мегабайт

Дан неориентированный граф  $G$  без петель и кратных рёбер, состоящий из  $n$  вершин и  $m$  рёбер. Вершины пронумерованы от 1 до  $n$ . Для каждой вершины  $1 \leq v \leq n$  скажите, сколько компонент связности будет в графе после её удаления.

### Формат входных данных

В первой строке даны два числа  $2 \leq n \leq 10^5$  и  $0 \leq m \leq 3 \cdot 10^5$  — количество вершин и рёбер в графе соответственно. В каждой из следующих  $m$  строк записаны по два числа  $1 \leq u, v \leq n$  — номера вершин концов соответствующего ребра.

### Формат выходных данных

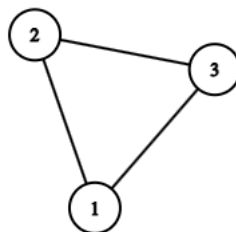
Выведите  $n$  строк. В строке номер  $1 \leq i \leq n$  должно быть написано одно число — количество компонент связности в графе после удаления из графа вершины номер  $i$ .

### Примеры

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

### Замечание

Граф из первого примера:



После удаления вершины номер 3 граф будет состоять из двух вершин и одного ребра:



## Задача 3F. [B'-B] T-Market efficiency team

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

Добро пожаловать в логистический центр гигантского маркетплейса T-Market! Чтобы оптимизировать затраты на доставку, была внедрена новая автоматизированная система: роботы-упаковщики должны объединять посылки в пары и отправлять их в одном контейнере. При этом формирование должно работать так, что в одной партии посылок все сформированные пары имеют одинаковый суммарный вес.

Вашей смене назначили на формирование  $N$  посылок для формирования контейнеров на отправку. Вас заверили, что партия идеальна сбалансирована, то есть из нее можно составить  $\frac{N}{2}$  пар посылок, так что каждая пара будет иметь одинаковый суммарный вес.

К сожалению, из-за сбоя в сортировочной ленте на вашу линию попали  $K$  лишних посылок из другого отдела. Теперь перед вашим роботом-манипулятором лежат  $N + K$  посылок, и ваша задача — запрограммировать его так, чтобы он отобрал ровно  $N$  посылок, из которых можно составить идеально сбалансированную партию, которую можно будет разбить на пары нужным образом, а  $K$  лишних оставил на ленте.

### Формат входных данных

В первой строке содержатся два целых числа  $N$  и  $K$  ( $1 \leq N \leq 150000$ ,  $0 \leq K \leq 400$ ,  $N$  — четное).

Во второй строке содержатся  $N + K$  различных целых чисел  $a_i$  ( $0 \leq a_i \leq 10^9$ ) — веса всех посылок, поступивших на вашу линию.

### Формат выходных данных

Выведите  $N$  целых чисел через пробел — веса посылок, которые образуют идеально сбалансированную партию. Числа нужно вывести в неубывающем порядке.

Если существует несколько решений, выведите любое из них.

### Пример

стандартный ввод	стандартный вывод
4 3 1 4 5 6 7 9 15	1 4 6 9

### Замечание

В примере из условия также верным ответом будет  $[4, 5, 6, 7]$ .

## Задача 4А. [В-Х] Покупка матрёшек

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

*Матрёшка* — традиционная русская игрушка в виде набора кукол, вложенных одна в другую. У каждой куклы есть размер (положительное целое число). Куклу размера  $x$  можно вложить в куклу размера  $y$  только если  $x < y$ .

На рынке расположено  $N$  ларьков, в каждом из которых продаётся ровно одна матрёшка. В ларьке  $i$  продаётся матрёшка размера  $a_i$ . Схема проходов между ларьками образует дерево (связный граф без циклов).

Турист очень торопится, поэтому действует следующим образом:

- Он выбирает какой-то ларёк  $k$  и идёт к нему от входа в рынок (ларёк с номером 1) по кратчайшему пути.
- В каждом ларьке на этом пути (включая 1 и  $k$ ) он может при желании купить матрёшку.
- Чтобы экономить время, турист сразу складывает матрёшки одну в другую. Поэтому всякая следующая купленная матрёшка должна быть строго больше всех предыдущих.

Как известно, чем больше вложенных матрёшек, тем интереснее сувенир. Найдите для каждого возможного выбора последнего ларька  $k$  максимальное количество матрёшек, которое турист сможет купить, следуя этим правилам.

### Формат входных данных

В первой строке входных данных дано одно целое число  $N$  — количество ларьков на рынке ( $2 \leq N \leq 2 \times 10^5$ ). Вторая строка содержит  $N$  чисел  $a_1, a_2, \dots, a_N$ , где  $a_i$  — размер матрёшки в  $i$ -м ларьке ( $1 \leq a_i \leq 10^9$ ).

В следующих  $N - 1$  строках описаны проходы между ларьками. Каждая строка состоит из двух целых чисел  $u$  и  $v$  ( $1 \leq u, v \leq N$ ,  $u \neq v$ ), обозначающих, что между ларьками  $u$  и  $v$  есть проход. Гарантируется, что эти проходы образуют дерево.

### Формат выходных данных

Выведите  $N$  строк. В строке с номером  $k$  выведите одно целое число — максимальное количество матрёшек, которое турист может купить, если его путь заканчивается в ларьке с номером  $k$ .

### Примеры

стандартный ввод	стандартный вывод
4 1 2 3 4 1 3 3 2 3 4	1 2 2 3
10 1 3 3 5 7 2 4 6 4 2 1 6 6 4 4 3 3 9 4 8 8 5 1 2 2 10 10 7	1 2 3 3 5 2 3 4 4 2

## Замечание

В первом примере

1. На пути до ларька 1 оптимально купить матрёшку размера 1.
2. На пути до ларька 2 оптимально купить матрёшек размера 1, 3.
3. На пути до ларька 3 оптимально купить матрёшек размера 1, 3.
4. На пути до ларька 4 оптимально купить матрёшек размера 1, 3, 4.



## Задача 4В. [В-Х] Плоды

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

Лето уже подходит к концу, а это значит, что на юге уже созрели плодовые деревья! Косте нужно подумать о том, чего бы сочного и южного ему привезти домой после продолжительного отдыха. Выбор остановился на трёх видах плодов: яблоки, сливы и персики. Плоды каждого вида лежат в вёдрах, в каждом ведре лежат плоды только одного вида. Плоды из вёдер нельзя смешивать между собой, также нельзя добавлять или выкладывать плоды из вёдер.

В каждом ведре с яблоками ровно  $A$  яблок, в каждом ведре со сливами ровно  $B$  слив, в каждом ведре с персиками ровно  $C$  персиков. Урожай получился большим, поэтому считайте, что вёдер каждого вида неограниченно много. Костя хочет взять с собой ровно  $X$  плодов в сумме (плод — это одно яблоко, одна слива или один персик). Он очень любит все эти плоды, поэтому он хочет взять хотя бы по одному ведру каждого вида. Тем не менее, Костя не сможет взять более  $N$  вёдер каждого из трёх типов (то есть вёдер с яблоками должно быть не более  $N$ , вёдер со сливами тоже не более  $N$ , и вёдер с персиками не более  $N$ ).

Косте стало интересно, сколько есть различных способов набрать ровно  $X$  плодов в сумме, соблюдая все указанные выше правила. Два способа считаются различными, если есть вид плодов, такой что в одном способе количество плодов этого вида отличается от количества плодов этого вида в другом способе. Помогите Косте найти ответ на его вопрос.

### Формат входных данных

В единственной строке входных данных даны пять целых чисел через пробел:  $N, A, B, C, X$  ( $1 \leq N \leq 10^6, 1 \leq A, B, C \leq 10^9, 1 \leq X \leq 3 \cdot 10^{15}$ ) — ограничение на количество вёдер каждого типа, количество яблок, слив и персиков в вёдрах соответственно, и суммарное количество плодов.

### Формат выходных данных

Выведите одно число — количество различных способов набрать ровно  $X$  плодов, соблюдая все правила из условия.

### Примеры

стандартный ввод	стандартный вывод
5 3 1 5 15	3
1 1 1 1 1	0
100000 31415 92653 58979 1000000000	2896

### Замечание

В первом примере входных данных Костя может взять:

- 1 ведро с персиками, 1 ведро со сливами и 3 ведра с яблоками
- 1 ведро с персиками, 4 ведра со сливами и 2 ведра с яблоками
- 2 ведра с персиками, 2 ведра со сливами и 1 ведро с яблоками

## Задача 4С. [В-Х] Бесконечные совпадения

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

Стартапер Вася придумал прекрасную идею для приложения для знакомств. Каждый пользователь будет вводить при регистрации свою любимую строку из маленьких латинских букв. Ниже обозначим за  $s \times k$  строку, состоящую из  $k$  повторений строки  $s$  (где  $s$  это какая-то строка из маленьких латинских букв, а  $k$  — натуральное число).

- Коэффициент привлекательности пользователя  $B$  (с любимой строкой  $t$ ) в глазах пользователя  $A$  (с любимой строкой  $s$ ) это максимальное число  $m$  такое, что  $t \times m$  является подстрокой  $s \times k$  для какого-то  $k$ . Если множество таких  $m$  неограничено, то коэффициент считается равным бесконечности.

Для дальнейшего построения рекомендательной системы Васе нужно эффективно вычислять коэффициенты привлекательности. Помогите ему это сделать для двух конкретных пользователей  $A$  и  $B$ .

### Формат входных данных

В первой строке входных данных вводится  $s$ , во второй строке вводится  $t$  — любимые строки пользователей  $A$  и  $B$ , соответственно ( $1 \leq |s|, |t| \leq 5 \times 10^5$ ), они состоят из маленьких латинских букв.

### Формат выходных данных

В единственной строке выходных данных выведите коэффициент привлекательности пользователя  $B$  в глазах пользователя  $A$ , если он конечен, и  $-1$ , если он равен бесконечности.

### Примеры

стандартный ввод	стандартный вывод
abcxabcabc abc	3
bbbbbbbbb bbb	-1
tgen algocourses	0
perfectmatch match	1

## Задача 4D. [B-X] Очередная задача на дерево

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

Преподаватели устали придумывать красивые легенды, поэтому в этой задаче формулировка будет формальной.

Дано дерево с  $N$  вершинами ( $1 \leq N \leq 100000$ ).

Для каждого  $K$  от 1 до  $N - 1$  определите, можно ли разбить все рёбра дерева на непересекающиеся пути длины ровно  $K$ .

### Формат входных данных

Первая строка содержит целое число  $N$ . Каждая из следующих  $N - 1$  строк содержит по два целых числа  $a$  и  $b$ , описывающих ребро между вершинами  $a$  и  $b$  ( $1 \leq a, b \leq N$ ,  $a \neq b$ ).

### Формат выходных данных

Выведите битовую строку длины  $N - 1$ . Для каждого  $1 \leq K \leq N - 1$   $K$ -й символ строки (слева направо) должен быть равен:

- «1», если возможно разбиение рёбер дерева на пути длины ровно  $K$ ;
- «0», иначе.

### Пример

стандартный ввод	стандартный вывод
13 1 2 2 3 2 4 4 5 2 6 6 7 6 8 8 9 9 10 8 11 11 12 12 13	111000000000

### Замечание

Для данного дерева возможно разбиение при  $K = 1, 2, 3$ . В случае  $K = 3$  одно из возможных разбиений следующее:

$$13 - 12 - 11 - 8, \quad 10 - 9 - 8 - 6, \quad 7 - 6 - 2 - 3, \quad 5 - 4 - 2 - 1.$$



## Задача 4Е. [В-Х] Волшебные столбы

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

Однажды один могущественный волшебник наткнулся на долину с  $N$  волшебными столбами, расположенными в ряд. Давным-давно в одной древней книге он вычитал, что если расположить эти столбы по возрастанию высоты слева направо, то произойдёт нечто невероятное и прекрасное. Разумеется, такую возможность пропускать нельзя, и наш волшебник тут же решил действовать. Однако двигать волшебные столбы не так просто, как может показаться.

Все столбы в долине перед волшебником были разной высоты, поэтому нужное расположение единственно. Будем называть  $i$ -м по высоте столбом такой, что он  $i$ -й по возрастанию высоты (то есть  $i - 1$  столбов ниже его, а остальные  $N - i$  выше). Каждый столб можно сдвинуть только с помощью какого-то из трёх заклинаний:

- Заклинание  $A$  двигает выбранный столб в любое место в ряду по желанию волшебника.
- Заклинание  $B$  двигает выбранный столб так, что он оказывается самым левым в ряду.
- Заклинание  $C$  двигает выбранный столб так, что он оказывается самым правым в ряду.

Любое заклинание можно применять к любому столбу произвольное число раз (в том числе ноль) и в произвольном порядке. Но разные заклинания, применённые к разным столбам, могут стоить волшебнику разного количества маны. А именно, применение заклинания  $A$  к  $i$ -му по высоте столбу стоит  $A_i$  маны, заклинания  $B$  к  $i$ -му по высоте столбу стоит  $B_i$  маны, а заклинания  $C$  к  $i$ -му по высоте столбу стоит  $C_i$  маны.

Мана восполняется очень долго, а тратится очень быстро. Поэтому волшебник хочет потратить как можно меньше маны на переупорядочивание волшебных столбов. Помогите ему в этом непростом деле и посчитайте наименьшее количество маны, необходимое для расположения столбов по возрастанию высоты.

### Формат входных данных

В первой строке входных данных даётся натуральное число  $N$  ( $1 \leq N \leq 2 \cdot 10^5$ ) — количество столбов в долине.

Во второй строке находятся  $N$  натуральных чисел  $P_i$  — описание расположения столбов:  $i$ -й столб слева является  $P_i$ -м столбом по высоте ( $1 \leq i, P_i \leq N$ , все  $P_i$  различны)

В каждой из следующих  $N$  строк даются три натуральных числа через пробел. В  $i$ -й строке ( $1 \leq i \leq N$ ) находятся числа  $A_i, B_i, C_i$  в таком порядке — количество маны для применения заклинаний  $A, B$  и  $C$  соответственно к  $i$ -му по высоте столбу ( $1 \leq A_i, B_i, C_i \leq 10^9$ )

### Формат выходных данных

Выведите одно число — наименьшее количество маны для искомого расположения столбов.

## Примеры

стандартный ввод	стандартный вывод
3 3 1 2 9 3 5 8 6 4 9 4 6	6
6 2 6 5 3 4 1 10 8 16 30 2 10 10 17 8 11 27 22 8 6 5 15 29 2	15

## Замечание

В первом примере входных данных можно применить заклинание  $C$  к третьему по высоте столбу: это будет стоить 6 маны и столбы после этого будут стоять по возрастанию высоты.

Во втором примере входных данных для искомого расположения можно применить заклинания в таком порядке:

- применить заклинание  $B$  к первому по высоте столбу — это стоит 8 маны
- применить заклинание  $C$  к пятому по высоте столбу — это стоит 5 маны
- применить заклинание  $C$  к шестому по высоте столбу — это стоит 2 маны

## Задача 4F. [B-X] Отменённые рейсы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	1024 мегабайта

Боливия состоит из  $n$  городов, их соединяет  $n - 1$  авиарейс местной авиакомпании. Из любого города существует способ добраться в любой другой, то есть рейсы образуют дерево.

К сожалению, авиакомпания решила сократить свои расходы и отменить некоторые рейсы. В министерство транспорта уже поступило  $m$  различных жалоб, что участники не могут добраться от города  $u$  до города  $v$ , потому что на пути есть хотя бы 1 отменённый рейс.

Вам поручили узнать количество вариантов, соответствующих всем жалобам, какие рейсы могли быть отменены. Два варианта считаются различными, если существует рейс, отменённый в одном из них и всё ещё существующий в другом.

### Формат входных данных

В первой строке дано целое число  $n$  ( $2 \leq n \leq 50$ ) - количество городов.

В следующих  $n - 1$  строках описаны авиарейсы — в  $i$ -й строке даны два целых числа  $a_i, b_i$  ( $1 \leq a_i, b_i \leq n, a_i \neq b_i$ ) — города, соединённые  $i$ -м рейсом. Гарантируется, что рейсы образуют дерево.

В следующей строке дано целое число  $m$  ( $1 \leq m \leq 20$ ) — количество жалоб.

Следующие  $m$  строк содержат жалобы — в  $i$ -й строке даны два целых числа  $u_i, v_i$  ( $1 \leq u_i < v_i \leq n$ ). Гарантируется, что все жалобы различны — то есть при  $i \neq j$  верно хотя бы одно из условий:  $u_i \neq u_j$  или  $v_i \neq v_j$ .

### Формат выходных данных

Выведите одно целое число — количество возможных вариантов отмены авиарейсов.

## Примеры

стандартный ввод	стандартный вывод
3 1 2 2 3 1 1 3	3
2 1 2 1 1 2	1
5 1 2 3 2 3 4 5 3 3 1 3 2 4 2 5	9
8 1 2 2 3 4 3 2 5 6 3 6 7 8 6 5 2 7 3 5 1 6 2 8 7 8	62

## Задача 5А. [X] Банкомат

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

В стране, где живет Петя, деньги представлены  $n$  видами купюр с номиналами  $a_1, \dots, a_n$  рублей, соответственно. Купюры, для удобства расчетов, устроены таким образом, что выполняются следующие свойства:

- $a_1 = 1$
- $a_2$  делится на  $a_1$ ,  $a_3$  делится на  $a_2$ ,  $\dots$ ,  $a_n$  делится на  $a_{n-1}$ .
- Все купюры различны, это означает, что в том числе выполнено  $a_1 < a_2 < \dots < a_n$ .

Однажды Петя зашел в магазин и купил там продуктов на  $m$  рублей. Петя заплатил  $x$  ( $x \geq m$ ) рублей и использовал для этого наименьшее возможное количество купюр, затем кассирша отдала Пете сдачу  $(x - m)$  рублей, также используя минимальное количество купюр.

Петя заметил, что ни одна из тех купюр, что он давал кассирше, не совпала по номиналу с тем, что ему отдали в сдаче. Этот факт его позабавил, но, к сожалению, он забыл, сколько именно денег дал кассирше. Поэтому он и обратился к вам за помощью.

От вас требуется вычислить количество различных значений  $x$ , при которых могли произойти вышеописанные события.

### Формат входных данных

В первой строке указаны числа  $n$  и  $m$  ( $1 \leq n \leq 50$ ,  $1 \leq m \leq 10^{15}$ ) — количество различных видов купюр и сумму, на которую закупился Петя.

Во второй строке указаны числа  $a_1, \dots, a_n$  ( $1 \leq a_1 < \dots < a_n \leq 10^{15}$ ) — номиналы купюр.

### Формат выходных данных

Выведите единственное число — ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
3 24 1 5 10	3
5 5 1 2 4 8 16	8

### Замечание

Значение  $x$  в первом примере может равняться 24, 25 или 30:

- При  $x = 24$  Петя бы заплатил  $10 + 10 + 4$  рублей, а сдачи ему не вернули;
- При  $x = 25$  Петя бы заплатил  $10 + 10 + 5$  рублей и ему бы вернули сдачу 1 рубль одной купюрой;
- При  $x = 30$  Петя бы заплатил  $10 + 10 + 10$  рублей, и ему бы вернули сдачу  $5 + 1$  рублей.

Во втором примере подходят следующие значения  $x$ : 5, 6, 8, 9, 10, 16, 17, 18.

## Задача 5В. [X] Положим положительными

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 5 секунд  
Ограничение по памяти: 512 мегабайт

Дан массив, состоящий из  $N$  целых чисел  $A_1 \dots A_N$ . Вам нужно ответить на  $Q$  запросов трех типов:

- $1 \ l \ r \ x$  : заменить  $A_i$  на  $A_i + x$ , для всех  $i \in [l, r]$ , причем  $x > 0$
- $2 \ i \ x$  : заменить  $A_i$  на  $A_i - x$ , причем  $x > 0$
- $3 \ l \ r$  : найти сумму **положительных**  $A_i$  среди  $i \in [l, r]$ .

### Формат входных данных

В первой строке входных данных вводятся два целых числа  $N$  и  $Q$  ( $1 \leq N, Q \leq 10^6$ ).

В следующей строке вводятся  $N$  целых чисел  $A_i$  ( $|A_i| \leq 10^6$ ).

В каждой из следующих строк вводится запрос одного из трех возможных типов:

- $1 \ l \ r \ x$  ( $1 \leq l \leq r \leq n, 0 < x \leq 10^6$ )
- $2 \ i \ x$  ( $1 \leq i \leq n, 0 < x \leq 10^6$ )
- $3 \ l \ r$  ( $1 \leq l \leq r \leq n$ )

### Формат выходных данных

На каждый из запросов третьего типа выведите одно целое число — искомую сумму.

### Пример

стандартный ввод	стандартный вывод
5 8	0
-3 0 2 3 -5	2
3 1 2	11
3 1 3	9
1 1 3 4	6
3 1 3	11
3 3 5	
2 4 10	
3 3 5	
3 1 5	

## Задача 5С. [X] Одностороннее движение

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

В стране есть  $n$  городов и  $m$  двусторонних дорог, которые их соединяют. Города пронумерованы числами от 1 до  $n$ , а  $i$ -я дорога ( $1 \leq i \leq m$ ) соединяет пару городов с номерами  $a_i$  и  $b_i$ . Какие-то пары дорог могут соединять одну и ту же пару городов, а какие-то дороги и вовсе могут вести соединять город с самим собой.

Власти страны проанализировали статистику ДТП и поняли, что все дороги надо в срочном порядке сделать односторонними, но пока не решили, как ориентировать каждую из дорог.

Есть  $k$  маршрутов, по которым ходит почта. Поэтому для каждого  $i$  ( $1 \leq i \leq k$ ) после выбора ориентации для каждой из дорог должен существовать маршрут от города  $x_i$  до города  $y_i$ .

Оказалось так, что **есть способ** ориентировать дороги так, чтобы удовлетворить условия выше. Но планов ориентации дорог, возможно, существует несколько. Поэтому для каждой из дорог государство хочет определить, в какую сторону может быть ориентированна каждая из дорог.

### Формат входных данных

В первой строке указана пара чисел  $n$  и  $m$  ( $1 \leq n, m \leq 100\,000$ ) — количество городов и количество дорог.

В последующих  $m$  строках указаны пары  $a_i, b_i$  ( $1 \leq a_i, b_i \leq n$ ) — концы  $i$ -й дороги.

Далее указано число  $k$  ( $1 \leq k \leq 100\,000$ ) — количество почтовых маршрутов.

В следующих  $k$  строках указаны пары чисел  $x_i, y_i$  ( $1 \leq x_i, y_i \leq n$ ) — начало и конец  $i$ -го почтового маршрута.

### Формат выходных данных

Выведите строку, состоящую из  $m$  символов, где  $i$ -й символ указывает как должна быть ориентирована  $i$ -я дорога:

- «R» означает, что направление движения должно быть от города  $a_i$  к городу  $b_i$ ;
- «L» означает, что направление движения должно быть от города  $b_i$  к городу  $a_i$ ;
- «B» означает, что дорога может быть направлена как в одну, так и в другую сторону.

Статус каждой из дорог определяется независимо. То есть символ «B» в позиции  $i$  должен быть написан в том случае, если существует какой-то план ориентации дорог, где  $i$ -я дорога ориентирована одним способом и должен существовать план, где эта дорога ориентирована в другую сторону.

### Пример

стандартный ввод	стандартный вывод
5 6 1 2 1 2 4 3 2 3 1 3 5 1 2 4 5 1 3	B B R B B L

## Задача 5D. [X] Подсчет путей

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	1024 мегабайта

Дана квадратная таблица, состоящая из  $n$  строк и  $n$  столбцов. Каждая клетка таблицы имеет цвет — клетка в  $i$ -й сверху строке и  $j$ -м слева столбце имеет цвет  $a_{i,j}$ , такую клетку будем обозначать парой  $(i, j)$ .

Назовем последовательность из одной или более клеток  $(r_1, c_1), \dots, (r_k, c_k)$  хорошей, если первая и последняя клетки имеют одинаковый цвет (то есть  $a_{r_1, c_1} = a_{r_k, c_k}$ ), а также для любого  $i$  от 1 до  $k-1$ ,  $(i+1)$ -я клетка в последовательности является либо правой, либо нижней соседней клеткой по отношению к  $i$ -й клетке. То есть выполнено либо  $(r_{i+1}, c_{i+1}) = (r_i+1, c_i)$ , либо  $(r_{i+1}, c_{i+1}) = (r_i, c_i+1)$ .

Вычислите количество хороших последовательностей клеток, которые можно найти в таблице. Так как ответ может оказаться слишком большим, то выведите его остаток при делении на 998 244 353.

### Формат входных данных

В первой строке дано содержится число  $n$  ( $1 \leq n \leq 400$ ) — размер таблицы.

В следующих  $n$  строках содержится по  $n$  целых чисел. В  $i$ -й строке  $j$ -е число — это  $a_{i,j}$  ( $1 \leq a_{i,j} \leq n^2$ ), цвет соответствующей клетки.

### Формат выходных данных

Выведите единственное целое число — ответ на задачу по модулю 998 244 353.

### Пример

стандартный ввод	стандартный вывод
2 1 3 3 1	6

### Замечание

В указанном примере существует 6 хороших последовательностей:  $[(1, 1)]$ ,  $[(1, 2)]$ ,  $[(2, 1)]$ ,  $[(2, 2)]$ ,  $[(1, 1), (1, 2), (2, 2)]$ ,  $[(1, 1), (2, 1), (2, 2)]$ .



## Задача 5Е. [X] Все префиксы всех суффиксов

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

Наибольшим общим префиксом двух строк  $X$  и  $Y$  — это строка максимальной длины, являющаяся префиксом и  $X$ , и  $Y$ . Обозначим её как  $\text{LCP}(X, Y)$ . Например,  $\text{LCP}(\text{aba}, \text{abc}) = \text{ab}$ ,  $\text{LCP}(\text{x}, \text{xy}) = \text{x}$ , а  $\text{LCP}(\text{abc}, \text{xy}) = \varepsilon$  (где  $\varepsilon$  это пустая строка).

Дана строка  $T$  длины  $n$ , состоящая из маленьких латинских букв. Для каждого  $k$  ( $1 \leq k \leq n$ ) обозначим через  $T_k$  суффикс строки  $T$ , начинающийся с  $k$ -го символа. Пусть

$$S_k = \sum_{i=1}^n |\text{LCP}(T_k, T_i)|.$$

От вас требуется найти значения  $S_1, S_2, \dots, S_n$ .

### Формат входных данных

Первая строка входных данных содержит одно число  $n$  — длину строки  $T$  ( $1 \leq n \leq 10^6$ ). Вторая строка содержит строку  $T$  длины  $n$  из маленьких латинских букв.

### Формат выходных данных

Выведите  $n$  целых чисел  $S_1, \dots, S_n$ , каждое в отдельной строке.

### Примеры

стандартный ввод	стандартный вывод
3 хуу	3 3 2
14 tbanktgentbank	20 17 15 14 11 11 8 7 8 11 8 6 5 2

### Замечание

В первом примере суффиксы строки хуу это хуу, уу, у, выполнено

- $\text{LCP}(\text{хуу}, \text{хуу}) = \text{хуу}$
- $\text{LCP}(\text{хуу}, \text{уу}) = \varepsilon$
- $\text{LCP}(\text{хуу}, \text{у}) = \varepsilon$
- $\text{LCP}(\text{уу}, \text{уу}) = \text{уу}$
- $\text{LCP}(\text{уу}, \text{у}) = \text{у}$
- $\text{LCP}(\text{у}, \text{у}) = \text{у}$

## Задача 5F. [X] Онлайн друзья

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	1024 мегабайта

Москва только хорошеет с годами. Так в  $2025 + 1 + \frac{1}{2} + \frac{1}{4} + \dots$  году вдоль серой ветки построили бесконечное количество зданий, которые нумеруются всеми возможными целыми числами.

Хоть у студентов ВШЭ и большая стипендия, но вместе с бесконечным количеством домов в Москве появился бесконечный спрос на жилище. Поэтому студентам приходится снимать квартиры группами.

Всего есть  $n$  групп студентов, причем известно, что  $i$ -я группа состоит из  $c_i$  студентов, которые решили поселиться в доме  $x_i$ . Из-за бытовых особенностей каждая группа состоит из студентов одного пола  $t_i$ , для юношей выполняется  $t_i = 1$ , а для девушек —  $t_i = 2$ .

Каждая группа студентов также регулярно выходит на прогулки, в ходе которых посещает участки серой ветки из какой-то окрестности их дома, не отходя от своего дома на расстояние большее, чем  $v_i$ . Из-за этих прогулок студенты иногда знакомятся друг с другом. Будем считать, что студенты из группы  $i$  знакомы со студентами из группы  $j$ , если выполнено  $|x_i - x_j| < v_i + v_j$ .

Каждый студент-юноша решил найти себе ровно одну онлайн-подружку, которая тоже учится во ВШЭ и с которой он на данный момент **не знаком**. Студенты волнуются, что если у одной девушки появится сразу два онлайн-друга, то у нее не хватит времени на учебу, и она отчислится, поэтому они хотят избежать такой ситуации.

Вам требуется определить, смогут ли все студенты-юноши найти себе по онлайн-подружке так, чтобы никакая пара юношей не подружилась с одной и той же девушкой.

Обратите внимание, что если нет ни одного юноши, который учится в ВШЭ, то ответ на задачу «Yes».

### Формат входных данных

В первой строке вам дано число  $1 \leq t \leq 3 \cdot 10^5$  — количество тестовых наборов. Следующие строки содержат описания тестовых наборов.

В первой строке каждого набора содержится целое число  $n$  ( $1 \leq n \leq 3 \cdot 10^5$ ).

Во второй строке указаны числа  $x_1, \dots, x_n$  ( $-10^9 \leq x_1 < x_2 < \dots < x_n \leq 10^9$ ).

В третьей строке указаны числа  $c_1, c_2, \dots, c_n$  ( $1 \leq c_i \leq 10^9$ ).

В четвертой строке указаны числа  $v_1, \dots, v_n$  ( $1 \leq v_i \leq 10^9$ ).

В пятой строке указаны числа  $t_1, \dots, t_n$  ( $1 \leq t_i \leq 2$ ).

Гарантируется, что сумма  $n$  по всем тестовым наборам не превосходит  $3 \cdot 10^5$ .

### Формат выходных данных

Для каждого теста в отдельной строке выведите «Yes» (без кавычек), если каждый из юношей может найти себе по уникальной онлайн-подружке, и «No» (без кавычек) иначе.

## Пример

стандартный ввод	стандартный вывод
4	Yes
2	No
10 20	No
1000000000 1000000000	Yes
4 4	
1 2	
2	
10 20	
1000000000 1000000000	
6 6	
1 2	
4	
0 9 13 15	
2 1 3 2	
6 6 1 1	
1 2 2 1	
4	
0 9 13 15	
2 1 3 2	
6 4 1 1	
1 2 2 1	