

Задача 1PracticeC1. [С практика] Перемещение карточек

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

Перед вами выложены n карточек слева направо. i -я слева карточка имеет цвет c_i : 'R', если карточка красная, и 'W', если карточка белая.

Вы можете делать операции двух типов (любое количество раз, в любом порядке):

- Выбрать любую карточку и поменять ее цвет (белый на красный, красный на белый)
- Выбрать две (необязательно соседних) карточки и поменять их местами

Назовем ряд карточек некрасивым, если в нём есть красная карточка, идущая сразу после (правее) белой. Сколько минимум операций нужно совершить, чтобы ряд стал красивым?

Формат входных данных

На первой строке задано число n ($1 \leq n \leq 2 \cdot 10^5$) — количество карточек. На второй строке задана строка из n символов c_i (цвет i -й карточки), при этом c_i может быть либо символом 'R', либо символом 'W'.

Формат выходных данных

Выведите одно число — минимальное количество операций, чтобы сделать ряд красивым.

Примеры

стандартный ввод	стандартный вывод
2 WR	1
2 WW	0
4 RWRW	1

Замечание

В первом примере можно поменять местами первую и вторую карточку, получим красивый ряд.

Во втором примере ничего менять не нужно, ряд и так красивый.

В третьем примере можно перекрасить вторую карточку.

Задача 1PracticeC2. [С практика] Поиск ноутбуков

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

Карантин позади, а значит, наконец-то можно приступить к очным занятиям! Но вот незадача: несколько школьников пришли без ноутбуков. Егор вызвался решить проблему и отправился на поиски техники. Всего в здании n этажей. Ноутбуки для кружка хранятся на одном из них. К сожалению, карантин был слишком долгим, а здание слишком высокое, поэтому Егор напрочь забыл, на какой этаж ему нужно. Но он не растерялся и решил спросить у Полины! На этом повествование можно закончить, но Полина сейчас читает лекцию, и у нее есть время отвечать только «YES» или «NO». Егор с Полиной коммуницируют следующим образом:

- Егор перечисляет номера этажей.
- Если среди перечисленных этажей есть этаж с ноутбуками, Полина отвечает «YES». Иначе Полина отвечает «NO».
- Процесс повторяется, пока Егор не дойдет до лифта (как известно, в лифте Интернет не ловит).

Как только Егор окажется в лифте, он попросит вашей помощи. Пожалуйста, помогите Егору понять, на каких этажах могут храниться ноутбуки, опираясь на ответы Полины.

Формат входных данных

В первой строке содержится число $1 \leq n \leq 10^5$ — количество этажей в здании. Далее идут строки-перечисления этажей (разумеется, все номера этажей — целые числа от 1 до n), при этом за каждой такой строкой идет ответ Полины («YES» или «NO»). Наконец, в последней строке содержится слово «HELP», означающее, что Егор зашел в лифт и просит вашей помощи.

Формат выходных данных

Выведите через пробел все номера этажей (в порядке возрастания), на которых могут храниться ноутбуки.

Пример

стандартный ввод	стандартный вывод
8 5 6 7 8 NO 1 2 3 4 YES 1 3 YES HELP	1 3

Задача 1PracticeC3. [С практика] Очередь

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

На олимпиаду приехали делегации нескольких стран. Теперь их представители выстроились в длинную очередь, чтобы заселиться в отель. При этом представители из одной страны очень ценят друг друга и любят общаться между собой. В частности, если в очереди подряд оказалось 3 и более члена из делегации одной и той же страны, то вся эта группа (стоящая подряд в очереди) бросает очередь и идет обсуждать хоккей в более спокойной обстановке, хвост очереди при этом подтягивается ближе к голове. Из-за этого у каких-то людей могут возникнуть новые соседи по очереди из их же страны, тогда процесс может повториться. Так как на стойке регистрации завис компьютер, то иными способами люди очередь не покидают. Хозяин отеля попросил Вас оценить убытки от зависшей аппаратуры и посчитать, сколько всего людей покинет очередь. Гарантируется, что в начальный момент времени в очереди есть не более одной группы, которая сразу может покинуть очередь.

Формат входных данных

В первой строке входного файла дано единственное число n ($1 \leq n \leq 1000$) — начальное количество людей в очереди. В следующей строке даны n чисел a_i , a_i — идентификатор страны i -го участника, $0 \leq a_i \leq 9$.

Формат выходных данных

Выведите единственное число — сколько людей покинут очередь.

Пример

стандартный ввод	стандартный вывод
7 4 2 3 3 3 2 2	6

Задача 1PracticeC4. [С практика] Зачет

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

На зачете в параллели C были введены оригинальные правила оценки. Каждому ученику давался большой массив чисел, после чего ему надо было найти в этом массиве подмассив (несколько последовательных элементов массива) с максимальной XOR-суммой. Как всегда, есть люди, для которых главное — получить не ноль. Помогите ленивым школьникам и найдите максимальную длину подмассива, XOR-сумма которого строго положительна.

Формат входных данных

В первой строке дано количество чисел в массиве n ($1 \leq n \leq 10^5$).

Во второй строке записаны n элементов массива. Каждый элемент является целым числом в диапазоне от 0 до 10^5 .

Формат выходных данных

Выведите единственное число - максимальную длину подмассива, XOR-сумма которого строго положительна.

Пример

стандартный ввод	стандартный вывод
6 0 0 3 7 4 0	4

Задача 1TheoryC1. [С теория] Сортировка по цвету

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

Есть n пар элементов, где первый элемент пары является числом, а второй — одним из трёх цветов: красный, синий или желтый.

Эти пары элементов отсортированы по числу. Разработайте алгоритм с временной сложностью $O(n)$ (то есть количество операций, совершаемых алгоритмом, должно расти линейно с ростом n) для сортировки пар по цвету (сначала красный, потом синий, в конце желтый) таким образом, чтобы сохранить порядок по числам для пар с одинаковым цветом.

Например, если на вход алгоритма поступает последовательность пар (3, жёлтый), (3, красный), (4, жёлтый), (5, синий), то на выходе алгоритм должен выдать последовательность пар (3, красный), (5, синий), (3, жёлтый), (4, жёлтый).

Задача 1TheoryC2. [С теория] Фишки

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

Дана клетчатая доска размера $N \times N$ и K фишек. Назовем сторонами доски первую и последнюю строку доски и первый и последний столбец доски (то есть всего 4 стороны). Вам необходимо расположить фишки в клетках доски таким образом, чтобы выполнялись два условия:

- В каждой клетке, находящейся на одной из сторон доски, должна оказаться хотя бы одна фишка.
- Количество фишек на стороне с минимальным количеством фишек должно быть как можно больше.

Опишите, как следует располагать фишки на доске, чтобы максимизировать количество фишек на стороне с минимальным количеством фишек.

Замечание

В этом примере $N = 4$ и $K = 15$. Если расположить фишки как на картинке, то на всех сторонах, кроме нижней, будет по 5 фишек, а на нижней стороне — 6. Значит минимальное количество фишек на стороне равно пяти. Можно доказать, что ответ лучше получить нельзя.

2	1	1	1
1			1
1			1
1	2	1	2

Задача 2PracticeSVp1. [С-В' практика] Путешествие по матрице

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

Вам дана нижнетреугольная матрица из n строк (i -й ряд состоит ровно из i элементов, расположенных в столбцах с номерами от 1 до i). В каждой ячейке матрицы записана маленькая латинская буква. Также у вас есть пустая строка s .

Вы стоите в ячейке с координатами $(1, 1)$ (первая строка, первый столбец). На каждом ходу можно делать переход на одну ячейку ниже или на одну ячейку по диагонали вправо-вниз. То есть из ячейки с координатами (i, j) можно попасть в ячейки $(i + 1, j)$ или $(i + 1, j + 1)$.

Каждый раз, когда вы попадаете в новую ячейку, к строке s в конец приписывается буква, стоящая в этой ячейке. Найдите лексикографически минимальную строку, которую можно получить при переходе из первой строки матрицы в последнюю (n -ю).

Формат входных данных

В первой строке задано число n ($1 \leq n \leq 5 \cdot 10^3$).

Затем идёт n строк, в i -й находится строка из i символов, отвечающая первым i столбцам i -й строки матрицы.

Формат выходных данных

Выведите искомую строку.

Примеры

стандартный ввод	стандартный вывод
3 o nu mwb	onm
4 a bb czc defr	abcd

Задача 2PracticeCВр2. [С-В' практика] Планировка города

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

Вы — мэр города Лайнбурга. Как можно догадаться из названия, все дома в вашем городе расположены вдоль одной бесконечной в обе стороны прямой. Прямо сейчас перед вами стоит задача спроектировать новый район. Задача осложняется двумя факторами:

- Для строительства вам доступны только n позиций вдоль прямой, заданные своими координатами $x_1, \dots, x_n$.
- В своей предвыборной программе вы обещали построить m школ, которые также можно строить только в точках с координатами x_i . В точках, в которых не будут построены школы, будут построены дома (бюджет надо осваивать целиком!).

Поскольку вы грамотный политик, то хотите, чтобы вашим гражданам было комфортно добираться от дома до школы. Ясное дело, что жители Лайнбурга рациональны и будут ездить от дома до ближайшей школы, а если таких несколько, то до любой из них. Расстоянием между домом и школой в этой задаче считается абсолютная разница координат.

Таким образом, ваша задача состоит в том, чтобы спроектировать новый район Лайнбурга так, чтобы минимизировать максимальное расстояние между домом и ближайшей школой.

В этой задаче мы не просим вас предъявлять подходящее расположение школ, достаточно вывести искомый минимальный максимум расстояния между домом и ближайшей школой.

Формат входных данных

В первой строчке вводятся два целых числа: n m . Во второй строчке вводятся координаты, в которых можно строить: x_1 x_2 $\dots$ x_n

- $1 \leq m \leq n \leq 10^5$
- Все x_i целые.
- Все x_i различные.
- $0 \leq x_i \leq 10^9$

Формат выходных данных

Одно целое число — ответ на задачу.

Пример

стандартный ввод	стандартный вывод
10 4 1 7 9 2 10 15 17 24 20 21	3

Задача 2PracticeCВр3. [C-В' практика] Трудности перевода

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

Недавно Глеб начал изучать новый язык - эльфийский, в магазине он смог найти только словарь с английского на эльфийский. Чтобы он не отчаялся, вы согласились ему помочь, теперь ваша задача составить словарь с эсперанто на английский из имеющихся переводов.

Формат входных данных

На первой строке вводится число n - количество слов в словаре.

На каждой из следующих n строк вам дается слово и всего его переводы в формате `word - translation1, translation2 ... translationk`. Английские слов и их переводы даны в лексикографическом порядке.

Все слова состоят только из маленьких латинских букв, длина каждого слова не превосходит 15 символов. Общее количество слов на входе не превышает 100000

Формат выходных данных

Выведите словарь с эсперанто на английский. Также как и в изначальном словаре, слова должны идти в лексикографическом порядке, также для каждого слова его переводы также должны идти в лексикографическом порядке.

Пример

стандартный ввод	стандартный вывод
3 greet - empfangen, willkommen silicon - silicon welcome - willkommen	3 empfangen - greet silicon - silicon willkommen - greet, welcome

Задача 2PracticeCBr4. [С-В' практика] Дизайнерское решение

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

Егор хочет оформить свою гостиную в стиле постмодерн с элементами барокко. Дизайнер предложил ему оформить градиентное освещение – вставить в специальную нишу ровно k лампочек так, чтобы они шли в одну линию, а интенсивность света строго увеличивалась слева направо. Чтобы эта линия смотрелась плавно и гармонично, необходимо чтобы мощности лампочек увеличивались строго на единицу при переходе от одной лампочки к другой слева направо. Будем называть степенью освещенности помещения мощность последней лампочки в нише (то есть самой яркой).

Егор любит продумывать все варианты, поэтому ему стало интересно, сколько различных степеней освещенности можно получить, используя имеющиеся у него в запасе лампочки. Лампочки имеются двух видов — одни рассчитаны на определенную мощность, таких у Егора n штук, все их мощности известны, для i -й лампочки из этой категории она составляет w_i . Вторую часть лампочек в количестве m штук Егору подарил дизайнер, который и был автором всей этой затеи. Эти лампочки можно отрегулировать под любую целочисленную мощность от 1 до 10^6 .

Помогите Егору и найдите, сколько различных уровней освещенности можно получить.

Формат входных данных

В первой строке даны три числа — число лампочек у Егора n ($1 \leq n \leq 10^6$), число лампочек, подаренных дизайнером, m ($1 \leq m \leq 10^6$) и количество лампочек в нише k ($1 \leq k \leq 10^6$). Во второй строке даны мощности n лампочек, мощность i -й лампочки составляет w_i Ватт ($1 \leq w_i \leq 10^6$).

Формат выходных данных

Выведите единственное число — сколько различных уровней освещенности можно получить.

Пример

стандартный ввод	стандартный вывод
8 3 5 1 5 60 58 61 11 27 25	9

Замечание

В примере из условия возможны 9 следующих вариантов:

- 1 2 3 4 5
- 56 57 58 59 60
- 57 58 59 60 61
- 58 59 60 61 62
- 59 60 61 62 63
- 60 61 62 63 64
- 23 24 25 26 27
- 24 25 26 27 28
- 25 26 27 28 29

Задача 2 Theory C Вр1. [С-В' теория] Сумма α -функций

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

Обозначим за $\alpha(n)$ наибольший делитель числа n , который не делится на 2. Очевидно, что такое число всегда есть — например, 1 подходит под условие.

Дано два числа L и R , больших 0, обозначающие границы отрезка целых чисел ($L \leq R$). Опишите алгоритм, который позволит вычислить сумму α на всём данном отрезке, то есть, $\alpha(L) + \alpha(L+1) + \alpha(L+2) + \dots + \alpha(R-1) + \alpha(R)$.

Пусть числа L и R ограничены сверху числом N , тогда время работы алгоритма, как функция от N , должно быть как можно меньше.

Задача 2TheoryCBr2. [C-B' теория] Раскраска фломастеров

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

Недавно Глебу подарили n коробок фломастеров, в i -й коробке a_i фломастеров. К сожалению, обнаружилось, что для всех фломастеров нужно выбрать цвет самому. У Глеба есть любимые цвета, поэтому для каждого фломастера Глеб может выбрать только один из k любимых цветов. Более того, Глеб хочет, чтобы для каждого цвета и для любых двух коробок с фломастерами количества фломастеров этого цвета отличались бы не более, чем на один. Глеб просит вас о помощи, вам нужно сказать, возможно ли это и если да, то предложить ему подходящую покраску. Если же решения не существует, то вам нужно это доказать. Требуемая асимптотика: $O(\max_{i \in \{1 \dots n\}} a_i \times n)$

Задача 3PracticeBpB1. [B'-B практика] Игнорируйте отрезок

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

Дан ряд из N клеток и набор из M отрезков, i -й из них содержит клетки от l_i до r_i включительно.

Для каждого отрезка i посчитайте количество клеток, которые бы не покрывал ни один отрезок, если бы i -го отрезка не было в наборе.

Формат входных данных

Первая строка содержит два целых числа N и M , $1 \leq N, M \leq 2 \cdot 10^5$. Последующие M строк содержат по два целых числа l_i и r_i , $1 \leq l_i \leq r_i \leq N$.

Формат выходных данных

Выведите M строк, в i -й из них выведите количество клеток которые не будут покрыты ни одним отрезком, если бы i -го отрезка не было в наборе.

Пример

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

Замечание

Для лучшего понимания разберем пример из условия. Если бы в наборе не было первого отрезка, остались бы свободны клетки 1 и 2. Если бы в наборе не было второго отрезка, осталась бы свободна только клетка 5. Если бы в наборе не было третьего отрезка, все клетки были бы заняты.

Задача 3PracticeBpB2. [В'-В практика] Переработка пластика

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

Любимый напиток Андрея — «Швепси». В магазинах он продается в литровых пластиковых бутылках и пользуется большой популярностью.

К сожалению, завод по производству этого напитка переживает не лучшие времена. Из-за трудностей с поставками, была разработана технология, позволяющая **ровно** m пустых использованных бутылок из под «Швепси» переработать в **ровно** k пустых бутылок того же размера, готовых к заполнению и продаже. В стране, в которой живет Андрей, коллективный разум восторжествовал, поэтому все использованные бутылки сдаются на переработку.

Для того, чтобы понять, сколько рабочих придется сократить, вам было поручено спрогнозировать продажи «Швепси» в этом году. Для каждой из n партий пустых чистых бутылок вычислите, сколько литров напитка можно с помощью неё продать, учитывая возможность утилизации пластика.

Формат входных данных

В первой строке содержатся целые числа n, m, k ($1 \leq n \leq 10^3, 1 \leq k < m \leq 10^6$).

Во второй строке содержатся n целых чисел b_i — число пустых литровых бутылок в i -й партии ($1 \leq b_i \leq 10^6$).

Формат выходных данных

Для каждого b_i выведите число литров напитка, которое можно продать, используя бутылки из i -й партии.

Примеры

стандартный ввод	стандартный вывод
2 12 2 72 60	86 70
1 10 4 22	34

Замечание

Во втором примере оптимальная последовательность действий выглядит следующим образом: в двадцати бутылках можно продать 20 литров «Швепси», после чего преобразовать использованные бутылки в восемь чистых. Далее в десяти чистых бутылках можно продать ещё 10 литров «Швепси», преобразовав десять пустых бутылок в четыре чистые бутылки, в которых можно продать дополнительные 4 литра напитка. Таким образом, с помощью этой партии можно продать 34 литра «Швепси».

Задача 3PracticeBpB3. [B'-B практика] Природа преобразилась

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1.5 секунд
Ограничение по памяти:	256 мегабайт

Коронавирус успел внести свои коррективы буквально во все сферы жизни. Экология — не исключение: стоило крупным промышленным заводам прекратить деятельность, а людям уйти на карантин, как природа вокруг начала преобразаться. Во многие города начали возвращаться редкие птицы и животные, а общая экологическая ситуация на планете начала улучшаться.

Вот и у Глеба во дворе выросло дерево и не просто дерево, а дерево с хог-плодами. Как и любое другое дерево, хог-дерево содержит n ветвей и $n - 1$ ребро, на каждой ветви растет плод, изначально каждый плод имеет вкус 0. В ходе наблюдений за деревом Глеб заметил, что произошло m событий двух видов :

1) Глебу стало интересно и он захотел узнать, а какой суммарный вкус имеют все плоды на пути в дереве между плодами a и b , к несчастью Глеба интересует не просто сумма, а $\oplus$ -сумма, $\oplus$ - операция битового хог

2) У всех плодов на дереве вкус изменился на x , то есть если до события вкус i -го плода был a_i , то теперь он становится $a_i \oplus x$

Формат входных данных

В первой строке вводятся два числа : n — количество вершин в дереве и m — количество событий, которые произошли с деревом. ($1 \leq n, m \leq 10^6$)

На второй строке — $n - 1$ число, где i -е число — индекс p — предка вершины $i + 1$ ($1 \leq p \leq i - 1$, $n, m \leq 10^6$)

Затем следуют m событий, сначала дается t - тип запроса, а затем либо a, b - если требуется найти ответ на запрос первого типа, либо x , если запрос второго типа. ($x \leq 10^9, 1 \leq a, b \leq n$)

Формат выходных данных

Требуется вывести ответ для каждого запроса первого типа.

Пример

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

Задача 3PracticeBpB4. [B'-B практика] Урбанистика

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

Насмотревшись видео про урбанистику, глава муниципалитета Николай Иванович решил сделать свой округ красивым и удобным для жизни. Начать преобразования он решил с одной стороны главной улицы. На этой улице находится ровно k домов, пронумерованных от 1 до k , каждый из этих домов нуждается в капитальном ремонте. Эта улица является кольцевой, поэтому после k дома идёт первый.

У главы муниципалитета есть n знакомых директоров компаний по благоустройству города. Каждая из этих компаний готова поспособствовать улучшению жизни горожан.

В процессе благоустройства сотрудники этих компаний пытаются сделать район лучше, но при этом избежать однообразия среди домов. Из-за этого каждая из компаний предложила следующее условие: она готова благоустроить фиксированное количество домов и при этом эти дома должны иметь последовательные номера на улице. Для i -й компании обозначим это фиксированное число домов как c_i .

В целях безопасности горожан работы будут производиться разными компаниями в разные дни, то есть, если одна компания будет ремонтировать какие-то дома в один день, то в этот день никакая другая компания не сможет проводить свои работы.

Так как глава муниципалитета очень дорожит дружбой со всеми главами компаний, он решил, что каждая из компаний будет привлечена для благоустройства. Все сотрудники компаний апатичны и в связи с этим готовы ремонтировать как дома, нуждающиеся в ремонте, так и те дома, что были ранее отремонтированы другими компаниями. Несмотря на это, коллектив компании получает особое удовольствие, если дом, который они ремонтируют нуждался в ремонте, то есть, над ним не работали представители других компаний. Глава муниципалитета, будучи человеком справедливым и честным, хочет максимизировать количество таких домов для каждой компании.

Более формально, необходимо для каждой компании выбрать c_i последовательных домов, над которыми они будут работать, а также порядок, в котором компании будут осуществлять свои проекты. После этого глава муниципалитета хочет выбрать такое число m , чтобы оказалось, что каждая из компаний отремонтировала хотя бы m домов, нуждающихся в ремонте. Это число m и требуется максимизировать. Помогите найти такое максимальное m .

Формат входных данных

Первая строка входного файла содержит два целых числа n ($1 \leq n \leq 10^5$) и k ($1 \leq k \leq 10^9$). Следующая строка содержит n целых чисел — значения c_i ($1 \leq c_i \leq k$).

Формат выходных данных

Выведите максимальное возможное значение m .

Примеры

стандартный ввод	стандартный вывод
2 100 5 10	5
4 10 7 8 3 5	2

Замечание

В первом примере $m = 5$, ведь одна из компаний не будет ремонтировать больше 5 домов. Она будет работать первой и отремонтирует 5 домов, после чего еще 10 неотремонтированных домов будут предоставлены второй компании.

Во втором примере достичь $m = 2$ можно, например, так. Сначала третья компания ремонтирует дома 4, 5, 6. Затем четвёртая — дома 1, 2, 3, 4, 5. Затем вторая — дома с 1 по 8, а затем первая — 6, 7, 8, 9, 10, 1, 2.

Задача 3TheoryBrV1. [В'-В теория] Табличный парадокс

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

По данным числам N , M , x и y определите, можно ли сделать таблицу чисел из N строк и M столбцов такую, что сумма чисел в ней строго положительная, а у любого подпрямоугольника таблицы размера $x \times y$, то есть состоящего из x строк и y столбцов, сумма чисел строго отрицательная.

Для случаев когда можно опишите алгоритм построения такой конструкции за $O(NM)$. Для случаев когда нельзя докажите это.

Не стесняйтесь использовать рисунки для лучшего объяснения конструкции.

Замечание

Для лучшего понимания приведем примеры. Например, при $N = M = 3, x = y = 2$ таблица может выглядеть так:

1	2	3
8	-30	4
7	6	5

С другой стороны, если $N = 2, M = 1, x = y = 1$ такую таблицу построить нельзя: оба числа должны быть отрицательными, но давать положительную сумму, что невозможно.

Задача 3TheoryBrV2. [B'-B теория] Наибольшая чередующаяся подстрока

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

Вам дана строка S , состоящая из нулей и единиц. Чередующейся подстрокой этой строки называется подстрока, имеющая вид либо $0101\dots$, либо $10101\dots$.

За одно действие вы можете разрезать строку в произвольном месте на две, развернуть каждую из них и склеить обратно в том же порядке. Например, если у вас была строка 0011010 , то вы можете разрезать её как $001 \mid 1010$, и получить строку 1000101 .

С помощью описанных выше операций вам нужно максимизировать длину наибольшей чередующейся подстроки. Операцию можно применять несколько раз, меняя место разреза строки.

Полным решением этой задачи будут считаться

- Алгоритм, решающий задачу за асимптотику $O(n)$.
- Доказательство того, что ответ вашего алгоритма действительно является наибольшим.

Задача 4PracticeBAr1. [B-A' практика] Карта

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

Маша и Саша нашли карту участка леса, разбитую на $n \times n$ квадратных участков со стороной 1 км. Некоторые из участков непроходимы. Ребята называют *маршрутом* строку, описывающую направления движения, каждое из которых обозначается одной буквой:

- «N» — пройти 1 км на север, то есть перейти из участка (x, y) в участок $(x, y + 1)$.
- «S» — пройти 1 км на юг, то есть перейти из участка (x, y) в участок $(x, y - 1)$.
- «W» — пройти 1 км на запад, то есть перейти из участка (x, y) в участок $(x - 1, y)$.
- «E» — пройти 1 км на восток, то есть перейти из участка (x, y) в участок $(x + 1, y)$.

Маша предложила Саше сыграть в игру. Она загадает начальный участок и строку, состоящую из символов «N», «S», «W» и «E», а Саша должен назвать количество подстрок, являющихся безопасными маршрутами. Маршрут безопасный, если при движении в соответствии с ним из изначальной позиции вы не встретите непроходимые участки и не выйдете за пределы карты.

Вам дана карта участка леса, а также начальный участок и строка, загаданные Машей. Найдите количество подстрок загаданной строки, являющихся безопасными маршрутами.

Формат входных данных

В первой строке входного файла даны два числа n и m ($1 \leq n \leq 1000$, $1 \leq m \leq 10^5$) — размер участка и длина загаданной строки.

Во второй строке входного файла дана строка s , состоящая из символов «N», «S», «W» и «E» длины m — загаданная строка.

Следующий n строк содержат по n символов в каждой и задают карту участка леса.

Символ «.» означает, что участок не содержит никаких препятствий. Символ «#» означает, что участок непроходимый. Символ «@» означает, что этот участок является начальным.

Гарантируется, что символ «@» встречается ровно один раз, а символ «#» — не более 300 раз.

В описании карты ось X направлена слева направо, а ось Y — снизу вверх.

Формат выходных данных

Выведите одно число — количество подстрок загаданной строки, являющихся безопасными маршрутами.

Пример

стандартный ввод	стандартный вывод
4 4 SESW .#.# .@#.#..	6

Замечание

Подстрокой (l, r) строки s называется непрерывная последовательность символов $s_l, s_{l+1}, \dots, s_r$, ($1 \leq l \leq r \leq |s|$).

В примере безопасными маршрутами являются следующие подстроки: $(1, 1)$: «S», $(1, 2)$: «SE», $(1, 3)$: «SES», $(3, 3)$: «S», $(3, 4)$: «SW», $(4, 4)$: «W».

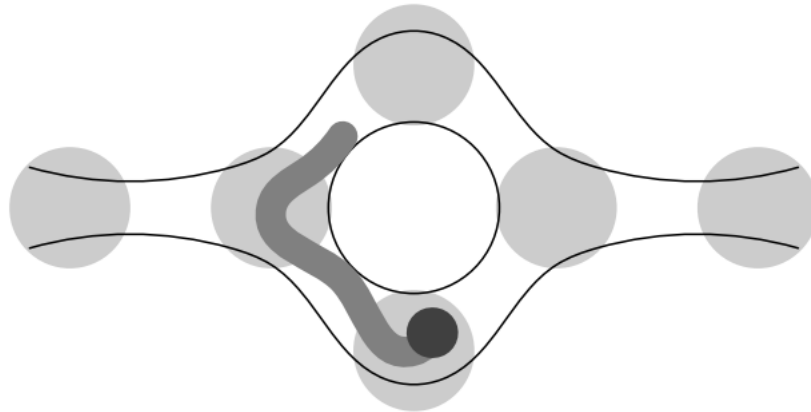
Задача 4PracticeVAr2. [B-A' практика] Шай-Хулуд

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 Шай-Хулуд. Они могут перемещаться по планете, по городам и дорогам, их соединяющим. Дорожная инфраструктура планеты очень специфична. Города связаны 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 каждый город лежит не более чем на одном простом цикле.

Червь ползает по дорогам, при этом невозможна ситуация, при которой две разные части тела червя, например, его голова и середина тела, находятся в одной точке. Вам известно начальное положение Шай-Хулуда, который может двигаться только головой вперёд. Также вам дана карта планеты. Ваша задача — вывести кратчайшее расстояние до указанного города.

Длина каждой из дорог равна ровно 1 км. Длина змеи равна $L - \frac{1}{2}$ км. Шай-Хулуд может переползать с дороги на дорогу, находясь головой в каком-то городе, а также его голова не может находиться в том же городе, что и другая часть его тела.



Формат входных данных

В первой строке ввода записано два числа n и k — количество городов и количество дорог на Арракисе ($1 \leq n, k \leq 100\,000$). В i -й из следующих k строк записано по два различных целых числа v и u — номера городов, соединенных i -й дорогой.

В следующей строке дано число L ($1 \leq L \leq k$), длина в километрах, на 500 м. большая настоящей длины червя. В следующей строке записано $L + 1$ число: номера городов, между которыми лежат последовательные сегменты Шай-Хулуда, начиная с головы. Между последней парой городов находится хвост Шай-Хулуда, занимающий половину дороги.

Исходное расположение червя корректно

В последней строке записано единственное целое число — город, в который песчаный червь очень хочет переместить свою голову.

Формат выходных данных

Если Шай-Хулуд может добраться до своей цели, выведите кратчайшую длину пути, то есть количество километров дорог, которое пройдёт голова червя. В противном случае выведите число «-1» (без кавычек).

Примеры

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

Задача 4PracticeVAr3. [B-A' практика] Поиск работы

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 512 мегабайт

Настя подается на стажировку в престижную компанию «Янкофф». На собеседовании ей досталась следующая сложная задача.

Дан массив положительных целых чисел $x_1, \dots, x_n$. Имеется q запросов одного из двух видов:

- Присвоить i -му элементу массива значение t .
- Посчитать количество делителей у произведения $x_l \cdot \dots \cdot x_r$. Так как это число может быть довольно большим, требуется найти его остаток по модулю $10^9 + 7$.

Помогите Насте решить эту задачу, иначе она никогда не найдет работу.

Формат входных данных

В первой строке дано число n ($1 \leq n \leq 5 \cdot 10^4$) — число элементов в массиве.

Во второй строке дано n чисел x_i ($1 \leq x_i \leq 10^4$) — элементы массива.

В третьей строке дано число q ($1 \leq q \leq 10^4$) — число запросов.

В следующих q строчках содержится по 3 целых числа. Если первое из этих чисел равно 0, то далее идут числа i и t ($1 \leq i \leq n$, $1 \leq t \leq 10^4$) для которых нужно выполнить первый запрос. В ином случае первое число равно 1, и далее идут числа l и r ($1 \leq l \leq r \leq n$) для которых нужно выполнить второй запрос.

Формат выходных данных

На каждый второй запрос выведите в отдельной строке количество делителей произведения $x_l \cdot \dots \cdot x_r$, взятое по модулю $10^9 + 7$.

Пример

стандартный ввод	стандартный вывод
5	12
2 5 6 3 7	12
6	16
1 3 5	12
1 2 4	
0 1 1	
0 4 7	
1 2 4	
1 3 5	

Задача 4PracticeVAp4. [B-A' практика] Ваня становится спортивным программистом

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.4 секунд
Ограничение по памяти: 256 мегабайт

Из-за коронавируса Ваня вынужден проводить все свое время дома. Он решил потратить это время с пользой — стать очень крутым спортивным программистом.

Пока он только выучил язык программирования, но не умеет хорошо решать задачи. Прочитав много условий задач он понял, какие объекты чаще всего встречаются в задачах по спортивному программированию. Он составил целый список таких объектов:

- массив целых чисел
- разделение массива на m отрезков
- сумма чисел на отрезке
- побитовый and
- максимизировать

Теперь он понял, что если он решит все возможные задачи с комбинациями этих объектов, то он точно станет очень крутым спортивным программистом.

Первая комбинация объектов, из которых родилась задача, следующая:

Вам даны числа n и m и массив из n положительных целых чисел $a_1, a_2, \dots, a_n$. Разделите этот массив на m непустых отрезков так, чтобы побитовый and сумм на отрезках (m чисел) был как можно больше. Вам нужно найти этот максимальный возможный побитовый and.

Эту задачу он решить не смог, попробуйте вы!

Формат входных данных

В первой строке находится два целых числа n, m ($1 \leq m \leq n \leq 50$) — количество чисел в массиве и количество отрезков.

Во второй строке находится n целых чисел $a_1, a_2, \dots, a_n$ ($1 \leq a_i < 2^{50}$) — элементы массива.

Формат выходных данных

Выведите единственное целое число — ответ на задачу Вани.

Пример

стандартный ввод	стандартный вывод
10 4 13 15 31 9 14 28 1 7 29 2	24

Замечание

В первом тесте можно добиться ответа следующим разбиением на 4 отрезка:

$$(13 + 15) \text{ and } (31) \text{ and } (9 + 14 + 28 + 1 + 7) \text{ and } (29 + 2) = 24$$

Задача 4TheoryBAp1. [B-A' теория] Ограниченный OR

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

Вам дан массив целых чисел $a_1, a_2, \dots, a_n$ ($0 \leq a_i < A$ для некоторого числа A , степени двойки, просто ограничения). Вы можете предполагать, что арифметические и битовые операции с числами $< A$ производятся за $O(1)$ и число $A = 2^{30}$.

Вам дано некоторое целое число $0 \leq k < A$. Найдите такой отрезок $[l, r]$ максимальной длины, что $a_l | a_{l+1} | \dots | a_r \leq k$. Здесь мы обозначаем как $|$ операцию побитового ИЛИ. Если ни одного такого отрезка не существует, сообщите об этом.

Асимптотика алгоритмов, которая будет оценена:

1. $O(n \log A)$ на время и память
2. $O(n \log n)$ на время и память
3. $O(n)$ на время и память

Задача считается полностью решенной, если алгоритм работает за $O(n)$ по времени и памяти.

Задача 4TheoryBAp2. [B-A' теория] Турнир

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

Имеется невзвешенный полный граф на n вершинах. Каждое его ребро ориентированно в некоторую сторону. Расстоянием от вершины u до вершины v называется минимальная длина пути из u в v , и бесконечность, если его не существует. Найдите вершину графа, максимальное расстояние от которой до других вершин минимально. Если таких несколько, то достаточно найти любую.

Задача считается полностью решенной, если алгоритм работает за $O(n^2)$ по времени и памяти.

Задача 5PracticeApA1. [А'-А практика] Странный агрегат

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

Костя — фанат популярной интернет-платформы AlgoExpress, которая занимается доставкой разных товаров в любую точку России. Ему пришла очередная посылка, в которой лежал подозрительно странный агрегат. Вот беда — Костя не помнит, когда его заказывал. И зачем он нужен, тоже не помнит.

Агрегат представлял собой плоский многоугольник, имеющий центр масс. Как у любого плоского многоугольника с центром масс, у агрегата существовали устойчивые положения. Устойчивым положением называется такое состояние агрегата, когда его можно поставить на стол, после чего он сохранит равновесие. Более формально, положение называется устойчивым, если у многоугольника существуют две такие различные точки, которые можно поставить на стол (горизонтальную прямую), так чтобы все остальные точки многоугольника лежали не ниже уровня стола, а проекция центра масс на стол попадала строго внутрь отрезка между двумя точками опоры. Два устойчивых положения, которые получаются поворотом многоугольника на одинаковый угол, считаются одинаковыми.

Костя решил, что попробует поставить странный агрегат в устойчивое положение, чтобы вспомнить, покупал ли он его раньше. Разумеется, у агрегата может быть несколько устойчивых положений, в зависимости от поворота. Поэтому Костя будет поворачивать агрегат вокруг центра масс и в итоге проверит их все. Помогите Косте понять, сколько именно устойчивых положений он в итоге просмотрит.

Формат входных данных

В первой строке входного файла дано целое положительное число n и целые числа x_c, y_c ($3 \leq n \leq 1000$, $-1000 \leq x_c, y_c \leq 1000$) — число вершин в многоугольнике и координаты центра масс агрегата.

В следующих n строках даны целые числа x_i, y_i ($-1000 \leq x_i, y_i \leq 1000$) — координаты многоугольника в порядке обхода против часовой стрелки. Гарантируется, что многоугольник не содержит самопересечений.

Формат выходных данных

Выведите одно число — количество устойчивых положений агрегата.

Пример

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

Замечание

В примере из условия устойчивые положения задаются следующими парами точек:

1. $(0, 0), (2, 0)$
2. $(3, 1), (1, 3)$
3. $(0, 2), (0, 0)$

Задача 5PracticeApA2. [А'-А практика] Распространение вируса

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

Вы решили заразить весь мир разработанным вами вирусом.

Всего в мире есть n стран, пронумерованных целыми числами от 1 до n . Вы хотите придумать некоторый порядок стран $p_1, p_2, \dots, p_n$ в котором они будут заражаться. Это перестановка целых чисел от 1 до n .

Вы не хотите чтобы до того, как все страны заразятся, мир разработал вакцину, поэтому вы придумали следующее условие на порядок заражения, при котором этого точно не произойдет: для любой страны $1 \leq i \leq n$ все страны $1, 2, \dots, i-1$ должны заразиться либо до страны i , либо после страны i .

Для каждой пары стран $1 \leq i, j \leq n$ вы посчитали время a_{ij} необходимое для того, чтобы вирус из страны i перешел в страну j . Время, за которое весь мир заразится с помощью порядка заражения p равно $\sum_{i=1}^{n-1} a_{p_i p_{i+1}}$.

Конечно, вы бы хотели воплотить свой коварный план как можно быстрее, поэтому вы хотите найти минимальное время, необходимое, чтобы заразить все страны и при этом мир не смог бы разработать вакцину. Найдите это время.

Формат входных данных

В первой строке находится единственное целое число n ($2 \leq n \leq 1500$) — количество стран.

В следующих n строках содержится по n целых чисел. В i -й из этих строк находятся $a_{i1}, a_{i2}, \dots, a_{in}$ ($0 \leq a_{ij} \leq 1000$).

Гарантируется, что $a_{ij} = a_{ji}$ и $a_{ii} = 0$.

Формат выходных данных

Выведите единственное целое число — ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
4 0 1 1 10 1 0 10 1 1 10 0 10 10 1 10 0	3
5 0 2 3 10 7 2 0 5 2 3 3 5 0 4 5 10 2 4 0 3 7 3 5 3 0	10

Замечание

В первом тестовом случае можно рассмотреть порядок заражения стран $p = \{4, 2, 1, 3\}$

Задача 5PracticeApA3. [А'-А практика] Первоклассная задача

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

Петя еще не закончил свою учебу в первом классе, но уже усиленно готовится к участию в международной олимпиаде школьников по информатике. Тренер дал Пете сложную задачу для подготовки. Кратко объясним ее условие:

Функция f от двух цифр A и B равна модулю разности между ними. Например, $f(3, 5) = f(5, 3) = |3 - 5| = 2$.

Функция g от двух чисел A и B равна сумме значений функции f по цифрам A и B , которые относятся к одному разряду. Если одно число короче другого, то оно дополняется ведущими нулями. Например, $g(102, 38) = f(1, 0) + f(0, 3) + f(2, 8) = 10$.

Программа получает на вход два числа A и B . Ответом на задачу будет сумма значений функции g по всем парам различных чисел, которые лежат в промежутке $[A, B]$.

Например, если $A = 290$, а $B = 292$, то пары будут такими:

- $g(290, 291) = g(291, 290) = 1$
- $g(290, 292) = g(292, 290) = 2$
- $g(291, 292) = g(292, 291) = 1$

Таким образом, ответ будет равен $2 \cdot (1 + 2 + 1) = 8$.

Поскольку Петя еще не проходил трехзначные числа, то он никак не может справиться с этой задачей. Он обратился за помощью к вам. Помогите Пете решить задачу!

Формат входных данных

В первой строке входного файла содержатся два целых положительных числа A, B ($1 \leq A \leq B \leq 10^{50000}$).

Формат выходных данных

Выведите одно число — ответ на задачу. Поскольку ответ может быть очень большим, от вас требуется не само значение, а его остаток от деления на число $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
290 292	8
2 6	40

Замечание

Первый пример разобран в условии. Во втором примере отрезку принадлежат числа 2, 3, 4, 5, 6.

Поскольку эти числа состоят всего из одной цифры, то ответом будет $\sum_{a=2}^6 \sum_{b=2}^6 |a - b| = 40$

Задача 5PracticeApA4. [А'-А практика] Пляжный сезон

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

Егор поехал купаться на реку. На каждом берегу реки есть несколько пляжей, каждый из которых имеет свой рейтинг красоты.

Некоторые пары пляжей соединены мостами через реку, однако никакие два из них не находятся на одной стороне реки. Все мосты двусторонние.

Егор хочет получить максимальное количество удовольствия от своей поездки. Для этого он должен найти такую последовательность посещения пляжей, чтобы никакие два использованных моста не пересекались и сумма красоты посещённых мест как можно больше. Два моста (a, X) и (b, Y) не пересекаются, если $(a \leq b \text{ и } X \leq Y)$ или $(b \leq a \text{ и } Y \leq X)$. Также нельзя использовать один и тот же мост несколько раз.

Помогите Егору хорошо отдохнуть. Он может начать и закончить свою поездку в любом месте.

Формат входных данных

Первая строка входных данных содержит три целых числа n , m и p ($1 \leq n, m \leq 40\,000$, $0 \leq p \leq 100\,000$) — количество пляжей на левом берегу, на правом берегу и количество мостов соответственно.

В следующих n строках содержится по одному целому числу a_i ($0 \leq a_i \leq 40\,000$) — красота i -го пляжа на левом берегу реки.

В следующих m строках содержится по одному целому числу b_i ($0 \leq b_i \leq 40\,000$) — красота i -го пляжа на правом берегу реки.

В последних p строках содержится по два целых числа u и v ($1 \leq u \leq n$, $1 \leq v \leq m$) описывающие мост между пляжем u на левом берегу и v на правом. Между каждой парой пляжей существует не более одного моста.

Формат выходных данных

Выведите одно целое число — максимальную суммарную красоту поездки.

Пример

стандартный ввод	стандартный вывод
2 3 4 5 3 2 1 2 1 2 2 1 1 3 2 2	9

Задача 5 Theory АрА1. [А'-А теория] Построить перестановку

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

Перестановка размера n это массив из n чисел, где каждое целое число от 1 до n встречается ровно 1 раз. Вам требуется найти перестановку размера n , на которую наложено m ограничений. i -е из ограничений требует, чтобы все числа перестановки, индекс которых лежит на отрезке от a_i до b_i были меньше всех чисел перестановки, индекс которых лежит на отрезке от c_i до d_i . Обозначим за L сумму длин всех отрезков $[a_i, b_i]$ и $[c_i, d_i]$. От вас требуется найти перестановку длины n , удовлетворяющую всем ограничениям за $O(n + L)$.

Задача 5TheoryApA2. [А'-А теория] Путешествие по Манхэттену

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

Манхэттенское расстояние между точками (x_a, y_a) и (x_b, y_b) определяется как $|x_a - x_b| + |y_a - y_b|$. Например, $dist((2, 3), (5, -3)) = 3 + 6 = 9$.

Вам дана последовательность из n точек в двумерном пространстве, а также q запросов. Запросы бывают двух типов:

1. Update $i (x, y)$. Этот запрос должен изменить i -ю точку в последовательности на точку (x, y) .
2. Find $l r$. Ответом на этот запрос будет кратчайшее манхэттенское расстояние, которое нужно пройти, чтобы по очереди посетить все точки на позициях с l -й до r -й **кроме, возможно, одной**. Непосещенную точку можно выбирать самому. Можно посетить все точки.

Например, если последовательность равна $(0, 0)$, $(1, 100)$, $(2, 0)$, то оптимальным ответом на Find 1 3 будет 2.

Придумайте алгоритм, который решает эту задачу за $O(n + q \log n)$.

Задача 6PracticeA1. [А практика] Росточек 3

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

Азат играет в компьютерную игру «Росточек 3». Цель игры — сажать растения в космосе. Азат собирается посадить свой любимый розмарин.

Космическая база состоит из n отсеков, некоторые из которых соединены односторонними переходами. В первом отсеке находится жилой блок. Именно там начинается игра.

Азат хочет посадить как можно больше растений. Он должен начать в отсеке 1, пройти по нескольким отсекам (возможно несколько раз), и закончить в отсеке 1. При этом в каждом посещённом отсеке Азат может посадить не более одного ростка розмарина.

Система отсеков очень неудобная, однако Азат может воспользоваться читом. Ровно один раз он способен пройти по одному из переходов в обратном направлении.

Помогите Азату! Найдите наибольшее количество растений, которое он может посадить, чтобы в его маршруте было не больше одного перехода, пройденного в обратном направлении.

Формат входных данных

Первая строка входных данных содержит два целых числа n и m ($1 \leq n, m \leq 100\,000$) — количество отсеков и количество переходов соответственно.

В следующих m строках содержится по два целых числа u и v ($1 \leq u, v \leq n, u \neq v$) описывающие переход между отсеками u и v . Один и тот же переход не появится более чем один раз.

Формат выходных данных

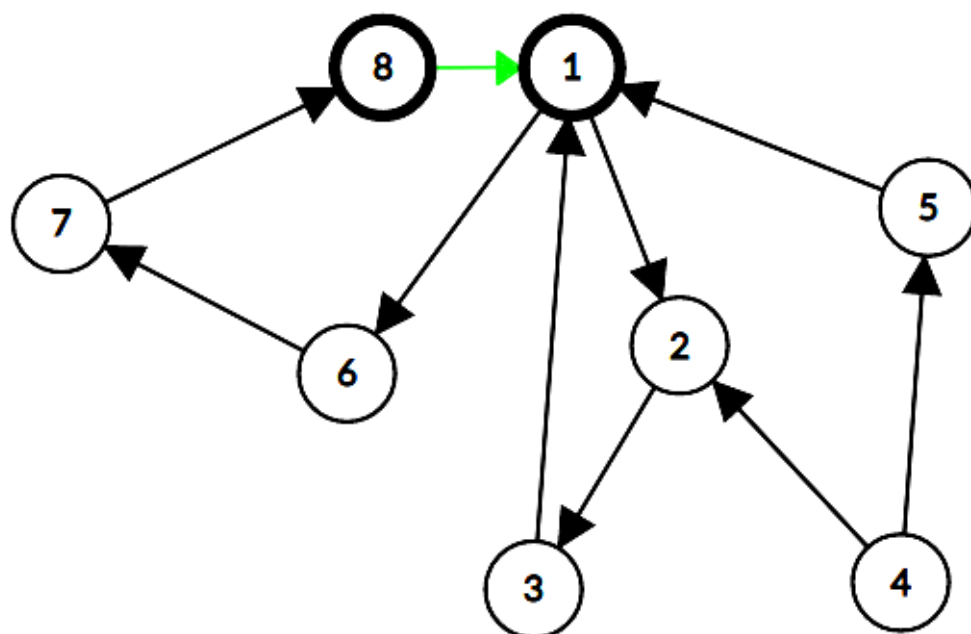
Выведите одно целое число — ответ на задачу.

Пример

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

Замечание

Пояснение к первому примеру:



Маршрут, который выбрал Азат: [1, 2, 3, 1, 6, 7, 8]

Также Азат мог бы воспользоваться маршрутом [1, 2, 3, 1, 2, 4, 5, 1], но в нём всего 5 различных отсеков.

Задача 6PracticeA2. [А практика] Захват склада

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	256 мегабайт

Вот уже несколько недель, как в Байтляндии идёт гражданская война между злостным диктатором, узурпировавшим власть, и мирными населением, желающим жить в свободе и демократии. Недавно случилось ужасное — войска диктатора захватили главный Байтляндский склад картошки, который содержит больше половины запасов еды в стране.

Чтобы не голодать, мирные жители решили отвоевать склад обратно. Но сделать это не так просто, ведь склад охраняется самыми бесчеловечными войсками диктатора, не дающими пощады даже женщинам и детям. Известно, что всего склад будут охранять n солдат, которые выстроятся в ряд. Так же солдаты будут разделены на батальоны размера m , причём первые m солдат в ряду будут принадлежать первому батальону, вторые m солдат будут принадлежать второму батальону и так далее. Разведке удалось установить, что солдаты будут совершать k перестроений двух типов:

1. Циклический сдвиг в каждом батальоне на a . Это означает, что если $a > 0$, то a раз последний солдат каждого батальона встаёт в начало батальона, а все остальные сдвигаются на 1 вправо, а если $a < 0$, то $|a|$ раз первый солдат батальона встаёт в конец батальона, а все остальные будут сдвигаться на 1 влево. После изменения 1 типа все солдаты остаются внутри своего батальона.
2. Циклический сдвиг всего ряда на a . Это означает, что если $a > 0$, то a раз последний солдат встаёт в начало ряда, а все остальные сдвигаются на 1 вправо, а если $a < 0$, то $|a|$ раз первый солдат встаёт в конец ряда, а все остальные будут сдвигаться на 1 влево. После изменений второго типа некоторые солдаты могут изменить свой батальон (так как меняется их порядковый номер в строю).

У каждого солдата есть уровень зверства, причём уровни зверства солдат остаются неизменными в процессе всех перестроений. Разведке мирных жителей удалось узнать, что после всех перестроений у первого солдата в ряду уровень зверства будет равен x_1 , у второго — x_2 , у третьего — x_3 , и так далее, у n -го солдата уровень зверства будет равен x_n . На основе этих данных, мирные жители хотят знать, какие будут уровни зверства у солдат в начальной расстановке. Помогите мирным жителям узнать эту информацию, ведь так они смогут найти самых добродушных солдат, подарить им цветы и переманить на свою сторону.

Формат входных данных

В первой строке вводятся 3 целых числа n , m и k ($1 \leq n, m, k \leq 100\,000$) — общее число солдат, число солдат в каждом батальоне и число перестроений. Гарантируется, что n делится нацело на m .

В следующих k строках описываются перестроения. В i -й из них вводятся два целых числа t_i и a_i ($1 \leq t_i \leq 2$, $-100\,000 \leq a_i \leq 100\,000$) — тип i -го перестроения и величина циклического сдвига i -го перестроения.

В последней строке вводятся n целых чисел $x_1, x_2, x_3, \dots, x_n$ ($0 \leq x_i \leq 100\,000$) — зверства солдат в порядке их следования после всех перестроений.

Формат выходных данных

В единственной строке выведите n чисел, i -е из них должно равняться зверству i -го солдата в начальном порядке следования.

Примеры

стандартный ввод	стандартный вывод
4 2 2 1 -1 2 3 1 2 3 4	1 4 3 2
12 3 3 1 2 2 5 1 -13 1 2 3 3 2 0 4 4 10 9 8 7	4 2 10 9 4 7 1 8 3 3 2 0

Замечание

В первом примере изначально зверства были 1 4 3 2, причём первый батальон содержал солдат со зверством 1 и 4, второй — со зверством 3 и 2. После первого перестроения зверства стали 4 1 2 3, а солдаты остались в своих батальонах. После второго перестроения зверства стали 1 2 3 4, причём первый батальон стал содержать солдат со зверствами 1 и 2, второй — со зверствами 3 и 4.

Задача 6PracticeA3. [А практика] Лаваш и кетчуп

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

Александр живёт в очень необычной стране. В этой стране есть ровно n городов, некоторые пары из которых соединены двусторонними дорогами. Каждая из них имеет целочисленную длину, причём между любыми двумя городами существует не больше одной дороги. Используя дороги можно добраться от любого города к любому другому городу.

В каждом городе готовят определённый вид шашлыка. Иногда в городе происходит переворот, и они меняют свой вид шашлыка на какой-то другой. После этой операции жители устраивают праздник.

Во время праздника в стране проходит популярное тв-шоу «Лаваш и кетчуп». Для того чтобы мероприятие было максимально интересным, два ведущих планируют посетить два города с различными видами шашлыка. Однако им будет грустно находиться далеко друг от друга, поэтому они хотят выбрать подходящие два города с минимальным расстоянием между ними. Гарантируется, что такие города найдутся.

Помогите им!

Формат входных данных

Первая строка входных данных содержит четыре целых числа n , m , d и q ($1 \leq n, m, q \leq 200\,000$, $1 \leq d \leq n$) — количество городов, количество дорог, количество видов шашлыка и количество запросов соответственно.

В следующих m строках содержится по три целых числа u , v и c ($1 \leq u, v \leq n$, $u \neq v$, $1 \leq c \leq 1\,000\,000$) описывающие дорогу между городами u и v , длиной c .

В следующей строке содержится n целых чисел a_i ($1 \leq a_i \leq d$) — начальные виды шашлыка в каждом городе.

В последних q строках содержится по два целых числа a и b ($1 \leq a \leq n$, $1 \leq b \leq d$), означающих, что в городе a теперь готовят шашлык вида b .

Формат выходных данных

После применения каждой операции выведите наименьшее расстояние между двумя городами с разными видами шашлыка.

Пример

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

Задача 6PracticeA4. [А практика] Робохоккей

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

На Олимпийские игры 2021 была предложена новая игра «Робохоккей». Правила игры оказались очень запутанными, поэтому вам предстоит в них разобраться.

Поле для игры представляет собой таблицу из n строк, пронумерованных числами от 1 до n сверху вниз, и m столбцов, пронумерованных числами от 1 до m слева направо. В некоторых клетках таблицы стоят роботы, каждый из которых смотрит в каком-то из четырех направлений.

Игра состоит из нескольких раундов. В каждом раунде каждый робот независимо от всех остальных выполняет следующий алгоритм:

1. Читает число x из своей памяти, относящееся к той клетке, в которой он сейчас находится (для разных роботов числа в одной и той же клетке могут отличаться).
2. Делает x поворотов на 90 градусов по часовой стрелке на месте.
3. Если после всех поворотов робот смотрит на клетку, находящуюся за пределами поля, то он поворачивается еще на 180 градусов.
4. Двигается на 1 клетку вперед.

При этом роботы достаточно маленькие, поэтому они никак не мешают друг другу перемещаться и даже могут стоять в одной клетке.

В одной из клеток таблицы находится финиш. Игра завершается, когда **все** роботы находятся на финише **одновременно**.

Вам необходимо посчитать, сколько раундов продлится игра.

Формат входных данных

Первая строка содержит высоту доски n ($3 \leq n \leq 50$), ширину доски m ($3 \leq m \leq 50$) и количество роботов k ($1 \leq k \leq 5$).

Вторая строка содержит координаты финиша x ($1 \leq x \leq n$) и y ($1 \leq y \leq m$).

Далее следуют описания роботов для каждого робота i от 1 до k :

Первая строка описания робота содержит его изначальные координаты x_i ($1 \leq x_i \leq n$) и y_i ($1 \leq y_i \leq m$), а также символ c_i , обозначающий направление, в котором изначально повернут i -й робот (L — налево, U — наверх, R — направо, D — вниз).

Каждая из следующих n строк описания робота содержит строку из m цифр, описывающих память i -го робота. Цифра a_{xy} ($0 \leq a_{xy} \leq 9$), находящаяся в x -й строке и y -м столбце, обозначает число, относящееся к клетке (x, y) в памяти робота i .

Для лучшего понимания формата входных данных смотрите пример.

Формат выходных данных

Выведите одно число — количество раундов в игре. Если игра никогда не закончится, выведите -1 .

Примеры

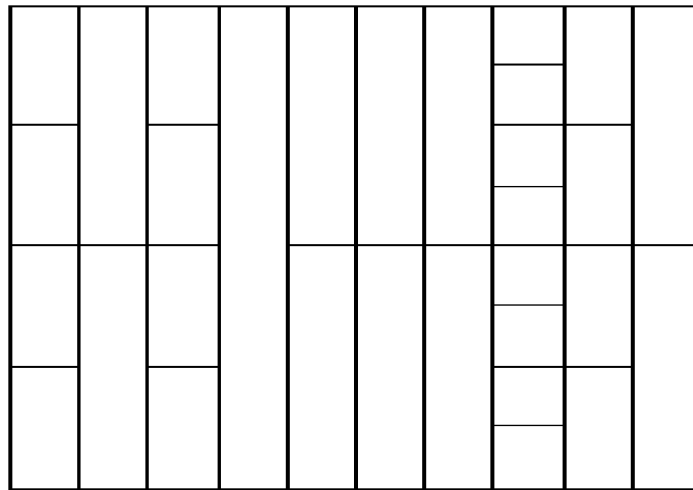
стандартный ввод	стандартный вывод
3 3 1 3 2 1 1 R 010 407 205	3
3 4 2 2 2 3 3 U 8324 3005 5112 3 1 R 4279 6101 0301	8

Задача 6TheoryA1. [А теория] Разрезание полосок

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

Дано n прямоугольников, ориентированных одинаково и расположенных в ряд слева-направо так, что их стороны примыкают друг к другу. (Смотрите иллюстрацию для более полного понимания.) Необходимо i -й из этих прямоугольников разрезать на a_i одинаковых частей, расположенных сверху-вниз. За одно действие можно сделать разрез на одном горизонтальном уровне сразу у нескольких *необязательно соседних* прямоугольников. Требуется за $O(\sum_{i=0}^n \sqrt{a_i})$ выяснить минимальное количество действий, чтобы сделать все необходимые разрезы.

Рассмотрим пример, когда $n = 10$ и $a_1, a_2, \dots, a_{10}$ равны 4, 2, 4, 1, 2, 2, 2, 8, 4, 2, соответственно. Вид прямоугольников после всех требуемых разрезов изображён ниже. Ясно, что минимальное количество действий, за которые можно добиться такого вида, равно 7.



Рассмотрим пример, когда $n = 3$ и a_1, a_2, a_3 равны 3, 4, 6, соответственно. В таком случае требовалось бы 7 действий.

Задача 6TheoryA2. [А теория] Изоморфизм деревьев

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

Дано дерево из n вершин, в котором степень каждой вершины не превосходит 4. Необходимо посчитать количество различных неизоморфных деревьев, которые можно получить добавлением к данному дереву одной вершины и одного ребра так, чтобы граф остался деревом, а степени всех вершин все еще не превосходили 4.

Временная сложность решения $O(n \log n)$.

Замечание

Два графа называются изоморфными, если существует такая биекция G между их вершинами, что вершины a и b соединены ребром тогда и только тогда, когда вершины $G(a)$ и $G(b)$ соединены ребром.