

Задача А. Автодополнение

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

Тор все-таки решил на покупку мобильного телефона, чтобы связываться со Мстителями было проще. В процессе написания смс-сообщений Тор не мог не заметить интересную и довольно полезную функцию — автодополнение. Эта функция по написанному непустому началу слова предлагает на выбор три самых популярных слова из своего словаря с таким же началом, чтобы для ускорения написания сообщения пользователь мог согласиться на одно из них, а не дописывать слово полностью (если таких слов меньше трех, предлагаются все возможные слова). Таким образом, пользователь при наборе сообщения может делать три действия — написать новую букву, удалить букву из конца написанного текста и согласиться на один из не более трех вариантов автодополнения.

Тони Старк любезно согласился взломать телефон Тора, чтобы узнать все слова из словаря, который использует функция автодополнения. Таких слов оказалось ровно n штук, а также оказалось, что автодополнение предлагает 3 самых первых слова из списка, то есть чем раньше слово находится в списке, тем популярнее оно считается. Тони также подметил, что система автодополнения устроена так, что если пользователь набрал слово s , и оно есть в словаре, то система не будет его предлагать.

Тор так заинтересовался автодополнением, что захотел узнать все способы набора своего сообщения s с помощью него. Понятно, что таких способов бесконечно много, поэтому Тор хочет найти все способы набора сообщения, используя не более k действий. Тор действует довольно логично и не собирается набирать новую букву так, что получившийся текст не будет являться префиксом желаемого сообщения s (однако он может согласиться на автодополнение, которое не будет являться префиксом s). Также после того, как сообщение s набрано, Тор может продолжить набор сообщения, если на текущий момент он сделал меньше k действий (а может и не продолжать и остановиться).

Для начала Тор решил ограничиться сообщениями, состоящими только из одного слова и найти не варианты набора сообщения, а только их количество по модулю $10^9 + 7$. Помогите ему с этой задачей — по данному слову s , состоящему из строчных латинских букв, и числу k найдите количество способов написать слово s не более чем за k действий.

Формат входных данных

В первой строке содержится число n — количество слов из словаря ($1 \leq n \leq 100$).

В следующих n строках содержатся слова w_i из словаря ($1 \leq |w_i| \leq 100$). Гарантируется, что каждое слово из словаря состоит только из строчных латинских букв, а также что суммарная длина слов из словаря не превышает 10^3 .

В $n + 2$ строке содержится строка s — слово, состоящее только из строчных латинских букв, которое хочет набрать Тор ($1 \leq |s| \leq 100$).

В последней строке содержится число k — максимальное количество действий (набор одного символа, удаление одного символа из конца текущего текста или соглашение на один из вариантов автодополнения), которое можно сделать ($1 \leq k \leq 10^3$).

Формат выходных данных

В единственной строке выведите количество способов набрать слово s по модулю $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
4 abacaba ababb abcabac babacb abacb 6	3
1 exactword exactword 4	7

Замечание

В первом примере возможны следующие три варианта написания слова «abacb»:

- 6 действий: набрать букву «a» → согласиться на автодополнение «abacaba» → удалить 3 раза последнюю букву → набрать букву «b»;
- 6 действий: набрать букву «a» → согласиться на автодополнение «ababb» → удалить 2 раза последнюю букву → набрать букву «c» → набрать букву «b»;
- 5 действий: набрать слово по одной букве.

Задача В. Локи и Шахматы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	7 секунд
Ограничение по памяти:	256 мегабайт

Однажды Тор заставил Локи играть в шахматы. Но шахматы в Асгарде не совсем обычные. Дело в том, что шахматная доска представляет собой таблицу из n строк и m столбцов. Каждая ячейка таблицы может либо иметь пешку, либо не иметь. Ячейка, которая не имеет пешки, называется свободной, а ячейка, которая имеет пешку, называется занятой.

Локи необходимо сделать q действий. Каждое действие заключается в том, что Тор дает Локи координаты ячейки и направление: 1 — вверх, 2 — вправо, 3 — вниз и 4 — влево. Если выбранная Тором ячейка свободна, то Локи ничего не должен делать, в противном случае он должен подвинуть пешку в указанном направлении.

Обратите внимание, что двигаются также все пешки, которые выбранная пешка толкает при движении в указанном направлении. Если пешка достигает границы доски, то дальше она не двигается.

Формат входных данных

В первой строке входных данных содержатся три числа n , m и q ($1 \leq n, m \leq 1000, 1 \leq q \leq 10^6$). Далее в n строках содержится строка из m символов. $c_{i,j}$ — j -й символ в i -й строке. Если $c_{i,j}$ равно единице, то в данной ячейке стоит пешка, в противном случае ячейка является свободной. В последних q строках содержатся три целых числа x , y и dir — строка и столбец запроса и направление ($0 \leq x \leq n - 1, 0 \leq y \leq m - 1, 1 \leq dir \leq 4$). Строки нумеруются сверху вниз от 0 до $n - 1$, столбцы нумеруются слева направо от 0 до $m - 1$.

Формат выходных данных

В n строках выходных данных выведите m символов. В i -й строке j -й символ должен быть равен единице, если после выполнения всех команд от Тора в текущей ячейке находится пешка; в противном случае, если пешки в текущей ячейке нет, выведите 0.

Пример

стандартный ввод	стандартный вывод
3 3 6	000
000	010
010	000
000	
1 1 2	
1 1 2	
1 2 3	
2 2 2	
2 2 4	
2 1 1	

Задача С. Побег с Асгарда

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

Тор и другие жители Асгарда бегут с него на корабле Грандмастера. Но сначала на этот корабль надо погрузиться.

Корабль Грандмастера состоит из двух палуб, причём на верхней палубе находится A посадочных мест, а на нижней палубе находится B посадочных мест. Вход на корабль находится на верхней палубе. Тор руководит посадкой жителей Асгарда на корабль. Но не всё так просто. Жители Асгарда (в том числе и Тор) распределились на группы друзей, причём они хотят, чтобы каждая группа находилась целиком на одной из двух палуб, чтобы им было удобнее общаться. Тор находится в первой группе. Так как Тор руководит посадкой на корабль, то он и его группа сядут на корабль последними, и, что бы не задерживать отправку корабля, хотят сесть на верхнюю палубу.

Помогите Тору, определите, возможно ли так разбить группы на две палубы, чтобы всем хватило места, и если возможно, то можно ли сделать так, чтобы группа номер 1, в которой находится Тор, села на верхнюю палубу.

Формат входных данных

В первой строке входных данных находится числа n , A и B — количество групп, на которое разбились жители Асгарда, количество посадочных мест на верхней палубе и количество посадочных мест на нижней палубе, соответственно ($1 \leq n \leq 100$, $1 \leq A, B \leq 10^5$). В следующей строке находится n целых чисел a_i — количество человек в каждой группе ($1 \leq a_i \leq 1000$).

Формат выходных данных

Если нельзя так разбить группы на две палубы, чтобы всем людям хватило мест, то выведите -1 . Иначе если можно так сделать, чтобы группа Тора ехала на верхней палубе, то выведите 1, а если нельзя так сделать, то выведите 2. В следующих двух строках выведите описание каждой палубы, сначала верхней, а потом нижней. Каждая палуба описывается числом групп, которое будут на этой палубе, а потом номерами этих групп. Группы занумерованы в том же порядке, что во входных данных, начиная с 1.

Пример

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

Задача D. Игра с числами

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

Локи заколдовал Тора и теперь Тор не может двигаться, пока не воспользуется контрзаклятием.

Для контрзаклятия требуется, чтобы уровень силы волшебного щита Тора был ровно $2^{60} - 1$. За одну секунду Тор может прибавить к уровню силы щита любое целое число в отрезке от a до b . В любой момент уровень силы щита — это целое число от -2^{60} до $2^{60} - 1$. Если при прибавлении выбранного Тором числа уровень силы щита выйдет из этих границ, то к его силе будет прибавляться 2^{61} до тех пор, пока его сила меньше -2^{60} , а затем из его силы будет вычитаться 2^{61} до тех пор, пока его сила больше $2^{60} - 1$.

Помогите Тору найти минимальное число секунд, через которое он сможет воспользоваться контрзаклятием.

Формат входных данных

В первой строке входного файла даны два целых числа a, b ($-2^{60} \leq a \leq 0 \leq b < 2^{60}$)

Формат выходных данных

Выведите единственное число - минимальное число секунд, через которое Тор сможет воспользоваться контрзаклятием, или -1 , если он не сможет освободиться никогда.

Примеры

стандартный ввод	стандартный вывод
-1 1	1152921504606846975
-2 1	576460752303423489

Задача Е. Защитники Асгарда

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

До Асгарда долетела весть, что богиня смерти, Хела, вырвалась из заточения. Защитникам Асгарда нужно срочно отретировать боевое построение. Структура войск в Асгарде выглядит следующим образом: войско состоит из n воинов, каждому воину присвоен уникальный номер от 1 до n , один из воинов является военачальником, у каждого воина есть от 0 до 7 подчиненных, каждый воин, кроме военачальника, является подчиненным ровно одного другого воина, военачальник не является ни чьим подчиненным, начиная с военачальника и переходя в подчиненного, можно прийти до любого воина. Иными словами, структура войска представляет собой подвешенное дерево, каждая из вершин которого имеет не более 7 детей.

Военачальник принял решение, что всех воинов нужно выстроить в шеренгу, при этом он хочет, чтобы в этой шеренге было как можно меньше инверсий. Инверсией называется пара воинов, таких, что номер воина идущего раньше больше, чем номер воина идущего позже. Построение происходит так: построение начинает военачальник, он упорядочивает своих подчиненных некоторым образом и в таком порядке вызывает их для совершения построения, очередной воин, вызванный для совершения построения, упорядочивает своих подчиненных некоторым образом, в таком порядке вызывает их для совершения построения, каждый раз перед вызовом следующего подчиненного дожидается, пока предыдущий завершит построение, затем встает в строй сам и завершает свое построение, в конце в строй встает военачальник. Обратите внимание, что стратегия поведения военачальника не отличается от стратегии поведения всех остальных воинов.

Вам дано описание армии Асгарда, найдите минимальное количество инверсий, которое может быть в построении, и определите, в каком порядке каждый из воинов должен вызывать своих подчиненных, чтобы достичь минимального количества инверсий в построении.

Формат входных данных

В первой строке даны два целых числа n и r — количество воинов в войске и номер воина, являющегося военачальником ($1 \leq n \leq 200\,000$, $1 \leq r \leq n$).

В следующих n строках даны описания подчиненных воинов. В i -й из них содержится список подчиненных i -го воина, он начинается с целого числа k_i — количества подчиненных i -го воина, далее следует k_i целых чисел c_{ij} — индексы воинов, являющихся подчиненными i -го воина ($0 \leq k_i \leq 7$, $1 \leq c_{ij} \leq n$).

Гарантируется, что каждый воин, кроме военачальника, появляется в списках ровно один раз, а военачальник не появляется ни в одном списке. Гарантируется, что списки задают дерево, подвешенное за вершину r .

Формат выходных данных

В первой строке выведите минимальное количество инверсий, которое может получиться в итоговом построении. В следующих n строках выведите описание войска в том же формате, что и во входных данных. Список подчиненных каждого воина выводите в том порядке, в котором он должен их вызывать на построение.

Если ответов несколько, можете вывести любой.

Примеры

стандартный ввод	стандартный вывод
4 2 0 3 3 1 4 0 0	2 0 3 1 3 4 0 0
7 2 2 7 6 3 1 5 3 1 4 0 0 0 0	11 2 6 7 3 3 5 1 1 4 0 0 0 0

Задача F. Тренировки Тора

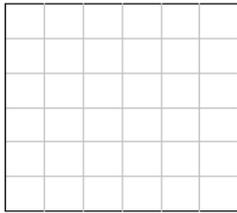
Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

После того, как Тор открыл в себе новые способности, позволяющие ему еще лучше повелевать молниями и громом, он решил тренировать эти навыки. Для тренировки он выбрал прямоугольное поле и мысленно разбил его на n равных строк и m равных столбцов, тем самым получив клетчатое поле $n \times m$. Разряд молний он решил направлять в виде контуров прямоугольников — для любых $1 \leq x_1 \leq x_2 \leq n$ и $1 \leq y_1 \leq y_2 \leq m$, он может поразить разрядом молнии клетки $(x_1, y_1), (x_1, y_1+1), \dots, (x_1, y_2), (x_1+1, y_2), \dots, (x_2, y_2), (x_2, y_2-1), \dots, (x_2, y_1), (x_2-1, y_1), \dots, (x_1+1, y_1)$.

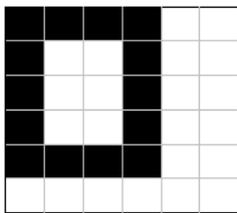
Чтобы лучше понимать, куда направить молнию, Тору после каждого разряда нужно знать, сколько связных областей образуют непораженные молнией клетки поля. Тор называет область связной, если любые ее две клетки достижимы друг из друга переходами по вертикали и горизонтали, а также если область максимальна по включению.

Например, события могут развиваться следующим образом:

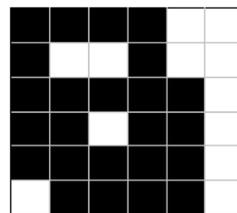
Сначала у Тора есть полностью непораженное поле 6×6 .



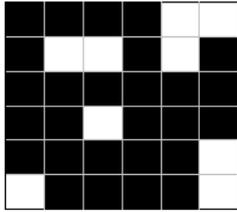
Затем Тор поражает разрядом молнии контур прямоугольника $(1, 1), (5, 4)$ и связанных областей становится 2.



Затем он поражает молнией контур прямоугольника $(3, 2), (6, 5)$ и связанных областей становится 4.



Финальным разрядом Тор поражает вырожденный прямоугольник $(2, 6), (4, 6)$ и количество связанных областей становится равным 5.



На вычисления руками у Тора нет времени, поэтому он попросил вас помочь ему с этой задачей.

Формат входных данных

В первой строке содержатся три числа n , m и q — размеры клетчатого поля и количество рядов молний, выпущенных Тором ($1 \leq n, m, q \leq 10^3$).

В следующих q строках содержатся описания рядов молний, выпущенных Тором. Описание состоит из 4 чисел x_1, y_1, x_2, y_2 — левого верхнего и правого нижнего углов прямоугольника ($1 \leq x_1 \leq x_2 \leq n, 1 \leq y_1 \leq y_2 \leq m$).

Формат выходных данных

В q строках выведите ответы на запросы — i -е число соответствует количеству связных областей, которые образуют непораженные молнией клетки, после i -го разряда молнии.

Пример

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

Задача G. Красивое число

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

Находясь в плену, Тор в свободное от битв время формировал свои вкусы в области чисел. В частности, он решил, что натуральное число является *красивым*, если все цифры его десятичной записи равны между собой.

Теперь Тор хочет научиться представлять любое натуральное число как сумму нескольких красивых (или, возможно, одного). Помогите ему в этом!

Формат входных данных

В единственной строке входных данных задано натуральное число $x < 10^{100}$. В числе x возможны ведущие нули.

Формат выходных данных

В первой строке выведите количество красивых чисел, сумма которых равна x . Во второй строке выведите сами эти числа через пробел.

Если возможных вариантов несколько, выведите любой из них.

Пример

стандартный ввод	стандартный вывод
13332	3 5555 5555 2222

Задача Н. Испытание

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

После схватки с Хелой Тор свалился на весьма странную планету Сакаар. Его сразу же захватили и отправили в качестве гладиатора на арену принимать участие в битве чемпионов Грандмастера, правителя планеты. Первое же испытание оказалось не физическим, а умственным, и показалось Тору чрезвычайно тяжелым.

Первоначально герою дали массив f из n чисел и поставили перед ним непосильную задачу: применить к массиву определенное количество раз запрашиваемые операции и после каждой операции ответить верно на вопрос, озвученный ниже. Операции бывают двух видов:

- OR x — к каждому элементу массива применить операцию побитового «ИЛИ» с числом x (то есть $f_i = f_i | x$ для всех $1 \leq i \leq n$).
- AND x — к каждому элементу массива применить операцию побитового «И» с числом x (то есть $f_i = f_i \& x$ для всех $1 \leq i \leq n$).

После каждой операции Тор должен назвать минимальное число непрерывных отрезков массива, на каждом из которых элементы расположены в неубывающем порядке.

Если вы желаете, чтобы Тор одержал победу в битве, помогите ему справиться с поставленной задачей. Только ваши усилия способны ему помочь!

Формат входных данных

Первая строка входных данных содержит натуральное число n — размер массива f ($1 \leq n \leq 10^5$).

Во второй строке находятся n целых чисел f_i — исходные элементы массива ($0 \leq f_i \leq 10^9$).

Третья строка содержит число m — количество операций ($1 \leq m \leq 10^5$). Следующие m строк содержат сами операции. Каждая строка содержит тип операции и целое число x , в формате, указанном в условии ($0 \leq x \leq 10^9$).

Формат выходных данных

Для каждой операции в отдельной строке выведите ее результат.

Пример

стандартный ввод	стандартный вывод
3	1
1 2 4	2
2	
OR 1	
AND 3	

Задача I. Гонки на колесницах

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	8 секунд
Ограничение по памяти:	256 мегабайт

Вслед за боями гладиаторов на арене, Грандмастер решил ввести второй вид спорта. Его выбор пал на гонки на роботизированных колесницах. Трасса для гонок содержит несколько перекрестков, соединенных прямыми участками трассы. Участки трассы не пересекаются, но могут иметь общие концы.

Тор прорабатывает маршрут, который позволит ему победить в гонке. Он выяснил, что гонка начинается на перекрестке v_1 , затем нужно посетить перекрестки с номерами v_2, \dots, v_k в таком порядке, и затем вернуться в v_1 . Перекресток может встречаться в списке несколько раз, но $v_i \neq v_{i+1}$ для любого $1 \leq i < k$, и $v_k \neq v_1$. В процессе поездки, колесница проезжает прямые участки с одинаковой постоянной скоростью, а в момент, когда приезжает на перекресток, мгновенно останавливается и начинает поворачиваться по направлению на следующий перекресток. В момент, когда колесница приезжает на перекресток, она ориентирована в направлении вектора из предыдущего перекрестка маршрута в текущий, затем она поворачивается по или против часовой стрелки, пока направление, в котором она ориентирована не станет совпадать с направлением на следующий перекресток маршрута. Когда колесница повернется по направлению на следующий перекресток, она продолжает движение.

Тор оценил, что колесница проезжает один метр за a секунд, и делает полный оборот вокруг оси за b секунд. Теперь он хочет узнать минимальное время, за которое можно завершить гонку. Начинать и заканчивать гонку можно будучи ориентированным в любом направлении.

Формат входных данных

В первой строке даны пять целых чисел n, m, k, a и b — количество перекрестков, количество участков трассы, соединяющих перекрестки, количество контрольных пунктов в маршруте, количество секунд, за которое колесница проезжает 1 метр, и количество секунд, за которое колесница делает полный оборот вокруг своей оси ($2 \leq n \leq 8\,000, 0 \leq m \leq n \cdot 3, 2 \leq k \leq 50, 1 \leq a, b \leq 10^9$).

В следующих n строках дано по два целых числа x_i и y_i — координаты i -го перекрестка в метрах ($-1\,000 \leq x, y \leq 1\,000$).

В следующих m строках даны по два целых числа a_i и b_i — номера перекрестков, которые соединяет i -й участок трассы ($1 \leq a_i, b_i \leq n, a_i \neq b_i$). Граф не содержит петель и кратных ребер. Гарантируется, что из любого перекрестка можно доехать до любого другого и что никакие два участка трассы не пересекаются, кроме как по концам. В том числе, конец участка трассы не может лежать на другом участке трассы.

В следующей строке даны k целых чисел v_i — номер перекрестка, на котором находится i -й контрольный пункт. Гарантируется, что никакие два подряд идущих контрольных пункта не находятся на одном перекрестке.

Формат выходных данных

В единственной строке выведите одно число — минимальное время, за которое можно завершить гонку. Выведите ответ с абсолютной или относительной погрешностью не более 10^{-6} .

Примеры

стандартный ввод	стандартный вывод
2 1 2 1 1 0 0 1 1 1 2 2 1	3.328427124746190
3 2 3 1 4 0 0 1 0 1 1 1 2 2 3 2 3 1	9.000000000000000

Задача J. Гармонический ряд

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 4 секунды
Ограничение по памяти: 16 мегабайт

Гармоническим рядом в математике называется следующая сумма:

$$\sum_{k=1}^{\infty} \frac{1}{k} = \frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots$$

Данный ряд обладает многими замечательными свойствами. Он расходится, а сумма первых n его членов стремится к $\ln(n) + \gamma$ (где γ — константа Эйлера — Маскерони) при стремлении n к бесконечности.

Тор не очень любит математический анализ, поэтому его не очень интересуют факты про замечательные свойства гармонического ряда. Зато Тор от скуки придумал новый математический объект и назвал его гармоническим рядом по простому модулю p .

Гармоническим рядом по простому модулю p Тор называет следующую сумму:

$$\sum_{i=1}^{p-1} \frac{1}{i} \pmod{p} = \left(\frac{1}{1} + \frac{1}{2} + \dots + \frac{1}{p-2} + \frac{1}{p-1} \right) \pmod{p}.$$

Числом $\frac{1}{x}$ для целого $x \in [1; p-1]$ по простому модулю p Тор называет такое целое число $y \in [1; p-1]$, что $x \cdot y \equiv 1 \pmod{p}$. Можно доказать, что если p — простое число, то для любого x , удовлетворяющего ограничению выше, найдётся ровно один подходящий y .

Тору стало интересно, чему равна сумма членов ряда от l до r , то есть чему равна сумма $\sum_{i=l}^r \frac{1}{i} \pmod{p}$.

Формат входных данных

В первой строке задано три целых числа l , r и p ($1 \leq l \leq r < p, r - l < 2 \cdot 10^7, p \leq 10^9$, p — простое).

Формат выходных данных

Выведите одно целое число — ответ на задачу. Обратите внимание, что все арифметические операции в данной задаче выполняются по модулю p .

Примеры

стандартный ввод	стандартный вывод
1 5 7	1
3 10 31	4

Замечание

Натуральное число x называется простым, если оно не равно 1 и у него нет делителей, отличных от 1 и x .