

## Задача 1. Новогодние подарки

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Как и земляне, жители Марса празднуют Новый год. У них есть и традиция дарить подарки на этот замечательный праздник.

У марсианского Деда Мороза есть  $x$  конфет, но он еще не решил, сколько из них достанется детям, а сколько останется его помощникам. Чтобы определить, сколько именно конфет достанется ребятам, дедушка раздал каждому из своих  $n$  помощников по одному целому числу  $c_i$  и решил действовать по следующему алгоритму:

1. Если у него остается строго меньше, чем  $y$  конфет, то выполнение алгоритма прерывается.
2. Иначе он просит помощников назвать одно из чисел  $c_i$  и число  $d$  такое, что  $0 \leq d \leq \lfloor \frac{x}{c_i} \rfloor$  (здесь  $\lfloor t \rfloor$  обозначает наибольшее целое число, не превосходящее  $t$ ). После этого Дед Мороз раздает  $d$  конфет детям, при этом оставшееся число конфет  $x$  уменьшается на  $d$ .
3. Если на предыдущем шаге было выбрано  $d = 0$ , то выполнение алгоритма прерывается, иначе алгоритм повторяется, начиная с шага 1.

После раздачи конфет ребятам все остальное достается помощникам.

Помощники Деда Мороза очень щедрые и, вообще говоря, не очень любят конфеты, поэтому они заинтересованы в том, чтобы как можно больше конфет досталось детям. Но помощники пока не знают, как им действовать оптимально, поэтому обратились за помощью к Вам.

Вам необходимо написать программу, которая находит, какое количество конфет останется помощникам Деда Мороза, если они будут стараться минимизировать это число.

### Формат входных данных

В первой строке входных данных находится три целых числа  $x$ ,  $y$  и  $n$  ( $1 \leq x, y \leq 2 \cdot 10^9$ ,  $1 \leq n \leq 10^5$ ) — изначальное количество конфет, параметр алгоритма Деда Мороза и количество помощников соответственно.

Во второй строке входных данных находится  $n$  целых чисел  $c_i$  ( $1 \leq c_i \leq 2 \cdot 10^9$ ) — число, выданное  $i$ -му помощнику.

### Формат выходных данных

Выведите одно целое число — количество конфет, которые останутся помощникам.

### Система оценки

№	Дополнительные ограничения	Баллы за подзадачу
1	$n = 1, 1 \leq x, y, c_i \leq 10^5$	48
2	$n \leq 100, 1 \leq x, y, c_i \leq 10^5$	23
3	$n \leq 1000$	10
4	Нет дополнительных ограничений	19

### Примеры

input.txt	output.txt
9 5 3 3 4 5	4
10 8 5 100 15 28 33 91	10
10 4 1 1	0

## Замечание

В первом примере последовательность действий может быть, например, такой:

1. Выбрать  $c_2 = 4$  и  $d = 1$  (это возможно, так как  $\lfloor \frac{9}{4} \rfloor = 2$ ). После этого остается  $9 - 1 = 8$  конфет.
2. Выбрать  $c_2 = 4$  и  $d = 2$  (учитывая, что  $\lfloor \frac{8}{4} \rfloor = 2$ ). Остается  $8 - 2 = 6$  конфет.
3. Выбрать  $c_1 = 3$  и  $d = 2$  (это снова возможно, поскольку  $\lfloor \frac{6}{3} \rfloor = 2$ ). После этого остается  $6 - 2 = 4$  конфеты.
4. Так как  $4 < y$ , то алгоритм заканчивается.

Существуют также и другие способы оставить четыре конфеты. Можно показать, что меньшее число конфет оставить нельзя.

Во втором примере, какое бы мы  $c_i$  ни взяли, все равно нельзя выбрать положительное значение  $d$ . Поэтому алгоритм остановится, и мы получим ответ, равный изначальному числу конфет, т. е. десяти.

В третьем примере можно взять  $c_1 = 1$  и  $d = 10$  и таким образом раздать детям все конфеты.

## Задача 2. Надежная сеть

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

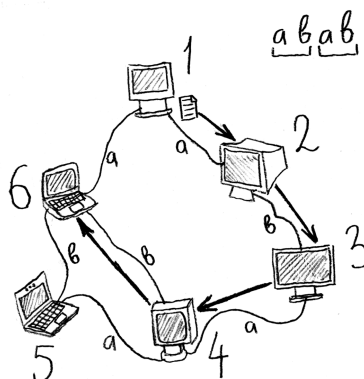
В компании «Interplanetary Software, Inc.» ведутся исследования, направленные на повышение надежности работы компьютерных сетей.

Инженеры компании построили сеть, состоящую из  $n$  компьютеров. Компьютеры соединены между собой  $m$  кабелями, по которым сигнал может ходить в обе стороны. Кабели бывают двух типов, которые обозначаются строчными латинскими буквами «a» и «b».

Компьютеры обмениваются информацией, посылая друг другу пакеты. В каждом пакете содержатся некоторые данные, а также *строка подтверждения*, изначально пустая. Каждый раз, когда пакет проходит по кабелю, то в конец строки подтверждения добавляется буква, соответствующая типу кабеля (если пакет прошел по a-кабелю, то добавляется буква «a», а если по b-кабелю — буква «b»).

Пусть два компьютера  $u$  и  $v$  обмениваются информацией. Тогда выбирается некоторый маршрут, проходящий через один или несколько компьютеров. Из-за ограничений конструкции сети маршрут не может проходить дважды через один и тот же компьютер. Затем по этому маршруту от  $u$  к  $v$  передается пакет. Получив пакет, компьютер  $v$  проверяет его строку подтверждения. Если эта строка непуста, ее длина четна и первая половина совпадает со второй, то тогда считается, что пакет доставлен надежно.

Рассмотрим рисунок ниже, на котором изображена сеть из шести компьютеров и семи кабелей.



На рисунке видно, что если выбрать маршрут  $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 6$  и отправить пакет по этому маршруту, то тогда строка подтверждения станет равна «abab». Ее длина четна, а правая половина («. . ab») совпадает с левой («ab. .»), т. е. пакет на компьютер номер 6 будет доставлен надежно.

Конечно же, построенная компанией «Interplanetary Software, Inc.» окажется бесполезной, если через нее нельзя ничего передать. Поэтому инженеры компании интересуются: существует ли такой маршрут между некоторой парой компьютеров, чтобы пакет, проходя по этому маршруту, был доставлен надежно?

### Формат входных данных

В первой строке входных данных находится два целых числа  $n$  и  $m$  ( $2 \leq n \leq 10^5$ ,  $1 \leq m \leq 2 \cdot 10^5$ ) — количество компьютеров и кабелей между ними.

В каждой из следующих  $m$  строк находится по два целых числа  $u_i$  и  $v_i$ , а также символ  $c_i$  ( $1 \leq u_i, v_i \leq n$ ,  $u_i \neq v_i$ ) — компьютеры, соединенные  $i$ -м кабелем, а также тип этого кабеля.

Гарантируется, что каждая пара компьютеров соединена напрямую не более чем одним кабелем.

### Формат выходных данных

Если требуемого маршрута не существует, выведите единственное целое число  $-1$ .

Иначе в первой строке выведите целое число  $k$  ( $2 \leq k \leq n$ ) — количество компьютеров в маршруте. Затем во второй строке выведите  $k$  целых чисел  $a_i$  ( $1 \leq a_i \leq n$ ) — компьютеры, через которые проходит требуемый маршрут в том же порядке, в каком они следуют в маршруте. Все компьютеры в маршруте должны быть различны.

Если существует несколько способов построить требуемый маршрут, выведите любой из них.

## Система оценки

№	Дополнительные ограничения	Баллы за подзадачу
1	$n \leq 200, m = n - 1, u_i = i, v_i = i + 1$	13
2	$n \leq 200, m = n - 1$ , сеть образует дерево	17
3	$n \leq 2000, m = n - 1, u_i = i, v_i = i + 1$	23
4	$n \leq 2000, m = n - 1$ , сеть образует дерево	24
5	Нет дополнительных ограничений	23

В подзадачах 2 и 4 дополнительное ограничение «сеть образует дерево» обозначает, что между каждой парой компьютеров существует ровно один маршрут.

## Примеры

input.txt	output.txt
6 7 1 2 a 2 3 b 3 4 a 4 5 a 5 6 b 6 1 a 4 6 b	5 1 2 3 4 6
4 3 1 2 a 2 3 b 3 4 a	-1
3 3 1 2 a 2 3 a 3 1 a	3 2 1 3
4 2 1 2 a 3 4 a	-1

## Замечание

Первый пример разобран в условии задачи.

Во втором примере мы никак не можем построить требуемый маршрут, поэтому необходимо вывести  $-1$ .

В третьем примере подходит любой маршрут длины два (например,  $2 \rightarrow 1 \rightarrow 3$ ).

В четвертом примере невозможно построить непустой маршрут четной длины, поскольку максимально возможная длина маршрута равна 1. Обратите также внимание, что в этом примере не существует маршрута между компьютерами 1 и 3, т. е. сеть необязательно должна быть связной.

### Задача 3. Столкновение галактик

Имя входного файла:	input.txt
Имя выходного файла:	output.txt
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Марсианские ученые активно занимаются исследованиями нашей Вселенной. Однажды группа ученых навела телескоп на участок неба, где активно образуются новые галактики.

Было обнаружено  $n$  галактик, каждая из которых содержала ровно одну звезду. Для каждой из этих звезд была известна их *яркость*. Если пронумеровать звезды целыми числами от 1 до  $n$ , то  $i$ -я звезда имеет яркость  $i$ .

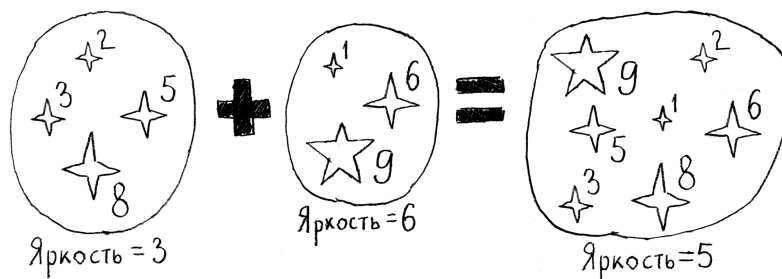
Во время наблюдения было замечено, что галактики время от времени сталкивались. При столкновении галактик образовывалась одна большая новая галактика, состоящая из всех звезд, которые находились в двух столкнувшихся галактиках. К примеру, пусть в первой галактике находились звезды  $\{1, 3, 5\}$ , во второй —  $\{2, 6\}$ , а в третьей —  $\{4, 7\}$ . Тогда после столкновения первой галактики со второй их останется всего две, причем первая будет содержать звезды  $\{1, 2, 3, 5, 6\}$ , а вторая —  $\{4, 7\}$ .

Время от времени ученые измеряли яркость галактик на небе. Эта величина равна медианной яркости звезды в галактике. То есть, если выписать яркости всех звезд галактики в порядке по возрастанию, то яркостью галактики будет значение элемента, который находится посередине в полученной последовательности. Если количество звезд в галактике четно, то из двух элементов посередине выбирается меньший. Например, если галактика содержит звезды с яркостью  $\{8, 5, 2, 3\}$ , то ее яркость равна 3. Действительно, если отсортировать яркости, мы получим  $\{2, 3, 5, 8\}$ . Из элементов посередине (3 и 5) мы выбираем меньший, т. е. 3.

Ученые проводили свои исследования довольно долго и накопили большой объем данных. Они не исключают возможность ошибки в своих наблюдениях, поэтому попросили написать программу, которая смоделирует поведение галактик. Вам требуется отвечать на два типа запросов:

- 1  $x y$  — галактика, содержащая звезду номер  $x$ , и галактика, содержащая звезду номер  $y$ , сталкиваются.
- 2  $x$  — найти яркость галактики, которая содержит звезду номер  $x$ .

Стоит отметить, что галактики сталкиваются настолько быстро, что яркость звезд никак не изменяется со временем.



#### Формат входных данных

В первой строке входных данных расположены два целых числа  $n$  и  $q$  ( $1 \leq n, q \leq 2 \cdot 10^5$ ) — количество звезд и количество запросов соответственно.

В каждой из следующих  $q$  строк находится по одному запросу. В начале строки находится целое число  $t_i$  ( $1 \leq t_i \leq 2$ ) — тип запроса. Далее расположено описание самого запроса:

- если  $t_i = 1$ , то далее следуют два целых числа  $x$  и  $y$  ( $1 \leq x, y \leq n, x \neq y$ ) — номера звезд, которые лежат в столкнувшихся галактиках. Гарантируется, что  $x$  и  $y$  находятся в разных галактиках.

- если  $t_i = 2$ , то далее следует целое число  $x$  ( $1 \leq x \leq n$ ), которое обозначает, что требуется вычислить яркость галактики, в которой находится звезда  $x$ .

## Формат выходных данных

Для каждого запроса второго типа выведите по одному числу в отдельной строке — ответы на запросы. Ответы должны идти в том же порядке, что и сами запросы во входных данных.

## Система оценки

№	Дополнительные ограничения	Баллы за подзадачу
1	$n, q \leq 1000$ , одна из сталкивающихся галактик состоит из одной звезды	8
2	$n, q \leq 2 \cdot 10^5$ , одна из сталкивающихся галактик состоит из одной звезды	25
3	$n \leq 1000$	17
4	$n \leq 8000$	9
5	Нет дополнительных ограничений	41

## Пример

input.txt	output.txt
9 13	2
1 2 8	4
1 3 5	3
2 8	6
2 4	5
1 8 3	5
2 2	7
1 1 6	
1 1 9	
2 1	
1 1 2	
2 8	
2 9	
2 7	

## Замечание

Рассмотрим пример.

После первых двух запросов образуются две новые галактики: одна содержит звезды  $\{2, 8\}$  и имеет яркость 2, а вторая содержит звезды  $\{3, 5\}$  и имеет яркость 3.

После пятого запроса образуется галактика  $\{2, 8, 3, 5\}$ . Если отсортировать звезды по возрастанию яркости, то получится  $\{2, 3, 5, 8\}$ , т. е. яркость галактики равна трем.

Седьмой и восьмой запросы образуют галактику  $\{1, 6, 9\}$ , которая имеет яркость 6.

Десятый запрос показан на рисунке выше. Полученная галактика содержит звезды  $\{2, 3, 5, 8, 1, 6, 9\}$ , что после сортировки дает  $\{1, 2, 3, 5, 6, 8, 9\}$ . Элемент посередине равен пяти, т. е. яркость галактики равна пяти.

## Задача 4. Города на Венере

Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

На планете Венера обитает немногочисленная, но очень развитая космическая цивилизация. Жители этой планеты построили  $n$  городов,  $i$ -й город имеет название  $a_i$ . У немногочисленной цивилизации и алфавит небольшой: жители Венеры используют только три буквы: «a», «b» и «c».

Два жителя Венеры (будем для простоты называть их Артем и Борис) решили сыграть в города. Правила игры такие. Сначала Артем называет некоторый город  $b_0$ . Затем Борис должен назвать какой-либо город  $b_1$ , который начинается на ту же букву, на которую заканчивается  $b_0$ . Далее Артем называет город  $b_2$ , который начинается на ту же букву, на которую заканчивается предыдущий город. Далее игра продолжается аналогичным образом, причем названные города не должны повторяться. Проигрывает тот, кто не может сделать ход.

Артем и Борис часто путешествуют вместе и во время долгих поездок часто играют в города. Настолько часто, что игра им надоела, и они стали искать оптимальную стратегию. Потратив на поиск стратегии не один день во время длительных поездок, друзья так ни к чему и не пришли. Поэтому они попросили помощи у Вас.

Вам требуется определить, кто выигрывает в города, если игроки действуют оптимально. Учитывайте, что Артем и Борис — опытные путешественники, поэтому знают названия всех  $n$  городов на Венере наизусть.



### Формат входных данных

В первой строке входных данных находится целое число  $n$  ( $1 \leq n \leq 10^5$ ) — количество городов на Венере.

В каждой из следующих  $n$  строк находится строка  $a_i$ , состоящая из строчных латинских букв «a», «b», «c» — название  $i$ -го города. Никакие два города не имеют одинаковое название.

Гарантируется, что суммарная длина всех названий городов не превосходит  $2 \cdot 10^6$  символов.

### Формат выходных данных

Если при оптимальной игре побеждает Артем, выведите «Artem» (без кавычек). Иначе выведите «Boris» (без кавычек).

### Система оценки

№	Дополнительные ограничения	Баллы за подзадачу
1	$n \leq 10$	7
2	$n \leq 17$	18
3	$n \leq 40$	19
4	$n \leq 300$	16
5	Названия всех городов не содержат букву «c»	26
6	Нет дополнительных ограничений	14

## Примеры

input.txt	output.txt
3 abacaba abaab acc	Artem
2 a aa	Boris
3 abbaab babbaa baba	Artem

## Замечание

В первом примере Артем может назвать, например, город «acc». Так как городов, начинающихся на «с», нет, то Борис не может сделать ход и проигрывает.

Во втором примере, если Артем назовет «aa», Борис назовет «a», и наоборот. После этого городов не останется, и Борис победит.

В третьем примере Артем называет «baba». Затем Борис вынужден ответить «abbaab». Далее Артем отвечает «babbaa» и выигрывает, поскольку города закончились, и Борис не сможет сделать ход.