

## Задача А. А как дела у остальных?

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

В этом году Чемпионат и Первенство России проводятся по экспериментальной схеме со штрафными попытками. Участники в таблице сортируются по количеству решённых задач (чем больше, тем лучше), а при равенстве — по штрафному времени (чем меньше, тем лучше).

Штрафное время — это сумма штрафных времён по сданным задачам. Штрафное время по сданной задаче определяется как минута с начала соревнования, на которой сдана задача (число от 1 до 180, так как соревнования идут 3 часа) плюс количество штрафных попыток по этой задаче, сделанных до того, как задача была сдана, умноженное на коэффициент (в этом году он равен 20).

В начале  $t$ -й минуты соревнования вы сдали все 9 предложенных задач, получив штрафное время  $p_0$ . Так как, в отличие от соревнований ICPC, в Чемпионате и в Первенстве России таблица не «замораживается», вы можете смотреть за результатами других участников до конца контеста.

Ваш коллега по региону на момент сдачи вами всех задач сдал  $s_1 < 9$  задач, текущее штрафное время у него равно  $p_1$ , а количество попыток по ещё не сданным задачам равно  $att$ .

Вам стало интересно, сможет ли он обогнать вас в таблице, если прямо сейчас сдаст все остальные задачи. Определите это по заданным значениям  $t$ ,  $p_0$ ,  $s_1$ ,  $p_1$  и  $att$ .

### Формат входных данных

Первая строка входных данных содержит одно целое число  $t$  — текущая минута с момента начала соревнований ( $1 \leq t \leq 180$ ). Вторая строка содержит одно целое число  $p_0$  — ваше штрафное время ( $200 \leq p \leq 10^4$ ). Третья строка содержит одно целое число  $s_1$  — количество решённых вашим коллегой задач ( $1 \leq s_1 \leq 8$ ). Четвёртая строка содержит одно целое число  $p_1$  — штрафное время вашего коллеги ( $200 \leq p \leq 10^4$ ). Пятая строка содержит одно целое число  $att$  ( $0 \leq att \leq 16$ ) — количество штрафных попыток по задачам, которые ваш коллега не сдал.

### Формат выходных данных

Если ваш коллега сможет вас обойти (то есть занять в таблице место строго выше вашего), выведите 0. Иначе (если он не сможет обойти, в том числе и в случае, когда он сможет только догнать) выведите 1.

### Примеры

стандартный ввод	стандартный вывод
100 501 8 400 0	0
80 502 7 322 1	1

## Задача В. Борьба с пожарами

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

В Стране Чудес есть  $n$  городов, пронумерованных целыми числами от 1 до  $n$ . Каждый город можно представить точкой на координатной плоскости. Из города  $i$  в город  $j$  телепортироваться можно тогда и только тогда, когда  $x_i - x_j \geq y_i - y_j$ , где  $x_i$  и  $y_i$  — координаты города  $i$ , а  $x_j$  и  $y_j$  — координаты города  $j$ .

Для размещения пожарных частей планируется найти такие города, из которых можно было бы телепортироваться в каждый город. Ваша задача — найти число таких городов и вывести их номера.

### Формат входных данных

Первая строка входных данных содержит одно целое число  $n$  — количество городов ( $2 \leq n \leq 10^5$ ).  $i$ -я из последующих  $n$  строчек задаёт координаты города  $i$  и содержит два целых числа  $x_i$  и  $y_i$  ( $-10^6 \leq x_i, y_i \leq 10^6$ ).

### Формат выходных данных

В первой строке выведите одно целое число  $k$  — количество городов, в которых можно разместить пожарные части. Последующие  $k$  строк должны содержать номера этих городов, по одному номеру на строку, отсортированные по возрастанию.

### Пример

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

## Задача C. Вместе со своим квадратом

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

Как многие из вас успели заметить, у Калининградской области код региона равен 39. Число 39 обладает следующим замечательным свойством: его десятичную запись можно разбить на две части так, что обе части задают корректные числа в десятичной записи (не содержащие ведущих нулей) и что вторая часть — это квадрат первой (разбиваем на 3 и 9,  $9 = 3^2$ ).

А, например, числа 42 или 101 таким свойством не обладают (в первом случае  $2 \neq 4^2$ , во втором при разбиении 1 01 вторая часть корректной десятичной записи не является, так как содержит ведущие нули), а при разбиении 10 1  $10^2 \neq 1$ .

Ваша задача — найти все числа с таким свойством между заданными  $\ell$  и  $r$  включительно.

### Формат входных данных

Первая строка входных данных содержит одно целое число  $t$  — количество тестовых примеров ( $1 \leq t \leq 10^4$ ).

Каждая из последующих  $t$  строк содержит по два целых числа  $\ell$  и  $r$  ( $1 \leq \ell \leq r < 10^{27}$ ).

### Формат выходных данных

Для каждого тестового примера выведите одно целое число — количество чисел с заданным свойством, расположенныхных между  $\ell$  и  $r$  включительно.

### Пример

стандартный ввод	стандартный вывод
2	2
23 39	0
1 10	

## Задача D. Главный инженер, магниты и шкафы

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

В Квантограде планируют построить новую железную дорогу на магнитной подушке, ведущую из Квантограда в Струногорск. Дорога идёт вдоль прямой, Квантоград находится в точке с координатой 0, Струногорск — в точке с координатой  $P$ .

В инфраструктуру дороги входят  $n$  мощных магнитов, пронумерованных от 1 до  $n$ .

Магнит с номером  $i$  определяется тремя параметрами — координатой  $x_i$ , силой притяжения  $a_i$  и радиусом действия  $r_i$ . Это означает, что магнит находится в точке с координатой  $x_i$ , и его сила притяжения  $a_i$  действует на все металлические конструкции с защитой  $a_i$  и меньше, которые находятся на расстоянии не более  $r_i$  от него.

Другими словами, если поставить металлическую конструкцию с защитой не более  $a_i$  на отрезке  $[x_i - r_i, x_i + r_i]$ , то на внутренности этой конструкции будет действовать поле магнита  $i$ .

Для работы железной дороги необходимо установить металлические релейные шкафы. Всего металлический завод Квантограда изготовил  $m$  разных металлических шкафов,  $j$ -й из этих шкафов имеет защиту  $b_j$ . Шкаф можно установить в любую точку с целой координатой от 0 до  $P$  включительно, если в этой точке не находится магнит или другой шкаф, и если эта точка не находится в радиусе действия хотя бы одного магнита, сила притяжения которого не меньше  $b_j$  — в противном случае оборудование будет повреждено магнитным полем. Если на точку действует несколько магнитов, то учитывается только тот из них, сила притяжения которого является наибольшей.

Главный инженер Квантоградских Железных Дорог Иннокентий хочет, чтобы при постройке дороги количество металлических шкафов была максимально возможным. Не пропадать же им зря!

Помогите Иннокентию ответить на вопрос, какое максимальное количество шкафов получится установить, а также определить, какие шкафы в какие точки нужно для этого поставить.

### Формат входных данных

В первой строке находятся два целых числа — количество магнитов  $n$ ,  $1 \leq n \leq 100\,000$  и  $P$ ,  $1 \leq P \leq 10^9$ , — координата Струногорска, конечной станции дороги.

В следующих  $n$  строках даны три числа через пробел: на каждой строке  $i$   $x_i$  ( $0 \leq x_i \leq P$ ),  $a_i$  ( $0 \leq a_i \leq 10^9$ ), и  $r_i$  ( $0 \leq r_i \leq 10^9$ ), — координата, сила и радиус действия соответственно для магнита  $i$ .

Далее следует строка, содержащая целое число  $m$ ,  $1 \leq m \leq 100\,000$  — количество металлических шкафов. Последняя строка файла содержит  $m$  целых чисел  $b_1, \dots, b_m$ ,  $1 \leq b_i \leq 10^9$  — защиты шкафов.

### Формат выходных данных

В первой строке выведите максимальное количество шкафов  $X$ , которое можно поставить. В следующих  $X$  строках выведите пары чисел — номер шкафа и координата, в которой он может быть установлен. Выводите шкафы в порядке возрастанию их номеров. Если ответов несколько, выведите любой из них.

## Примеры

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

## Замечание

В первом тесте первый шкаф можно поставить в точку с координатой 1. В ней действует магнит с нулевой точки, но его силы не хватает, чтобы пробить защиту первого шкафа. Второй шкаф можно поставить на координату 2. В ней не действует ни один магнит. В точку с координатой 3 получится поставить четвертый шкаф, а в точку с координатой 6 третий.

Во втором тесте не получится поставить ни один шкаф.

## Задача Е. Два игрока и кучи песка

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

Это интерактивная задача

Вам даны  $n$  куч песка, где  $i$ -я куча содержит  $a_i$  граммов песка.

Два игрока по очереди играют в игру. В каждом ходе игрок должен выбрать непустую кучу и удалить из нее произвольное положительное количество песка, выражющееся числом не более, чем с 9 знаками после десятичной точки. Если игрок не может сделать ход (т.е. в каких-либо кучах не осталось песка), он проигрывает игру. Ваша цель — либо **выиграть** игру, либо **продлить** ее достаточно долго.

### Протокол взаимодействия

Взаимодействие начинает программа жюри, выводя целое число  $n$  ( $1 \leq n \leq 1000$ ) — количество куч, а в следующей строке —  $n$  целых чисел  $a_i$  ( $1 \leq a_i \leq 1000$ ), указывающих размеры куч в граммах.

После этого начинается игра. Каждый ход представляет собой вывод двух чисел — целого числа  $x$  и вещественного  $v$  ( $1 \leq x \leq n$ ,  $0 < v \leq a_x$ ), указывающих номер выбранной кучи и вес удаляемого песка в граммах, соответственно. Значение  $v$  должно быть положительным вещественным числом с не более чем 9 знаками после запятой. Вы начинаете игру как первый игрок. Игроки делают ходы по очереди: сначала свои два числа выводите вы, затем — программа жюри и так далее.

Ваше решение будет признано верным, если либо программа жюри проигрывает, либо если ваша программа сделает 100 000 ходов без поражения.

Когда игра закончится, программа жюри выведет ‘0 0’ вместо хода. Получив этот сигнал, ваша программа должна корректно завершиться.

### Пример

стандартный ввод	стандартный вывод
3	
1 2 3	
2 1.500000000	1 0.000000001
3 2.500000000	2 0.5
3 0.100000000	3 0.3
3 0.050000000	3 0.05
0 0	1 0.999999999

### Замечание

Не забывайте после вывода очередного хода выводить символ перевода строки, а также сбрасывать буфер ввода-вывода с помощью вызова функции `flush` используемого вами языка программирования.

## Задача F. Есть ли такая буква?

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

У Кроша есть  $n$  строк  $s_1, s_2 \dots s_n$ , состоящих из строчных английских букв.

Обозначим через  $\ell$  суммарную длину всех строк.

Крошу нужно обработать  $m$  запросов, которые могут быть одного из двух типов:

- В запросе первого типа задаётся номер строки  $i$ ,  $1 \leq i \leq n$ , и пара букв  $c_1, c_2$ . Тогда Крош заменяет в строке под номером  $i$  все буквы  $c_1$  на буквы  $c_2$ . Например, если строка `abacabc`,  $c_1 = a$  и  $c_2 = b$ , то после замены Крош получит строку `bbbcbbc`.
- В запросе второго типа задаются два индекса  $i, j$  и буква  $c$  такие, что  $1 \leq i \leq j \leq \ell$ . Крошу нужно ответить, сколько раз буква  $c$  встречается на отрезке  $[i, j]$  в строке, полученной склейкой всех строк. То есть, если  $w = s_1 + s_2 + \dots + s_n$  (Обратите внимание, что  $s_i$  здесь — текущие версии строк, которые могут отличаться от изначальных), то ему нужно посчитать количество таких  $i \leq pos \leq j$ , что  $w_{pos} = c$ , где  $w_{pos}$  — символ, стоящий на позиции  $pos$  в строке  $w$ .

Помогите Крошу обработать запросы: для каждого запроса второго типа выведите ответ на него. Ответы на запросы выводите в том порядке, в котором они встречаются во входных данных.

### Формат входных данных

В первой строке содержатся два целых числа  $n$  ( $1 \leq n \leq 2 \cdot 10^5$ ) — количество строк и  $m$  ( $1 \leq m \leq 2 \cdot 10^5$ ) — количество запросов.

Далее идут строки  $s_i$ , по одной на строку. Все  $s_i$  состоят из строчных латинских букв, их суммарная длина  $\ell$  не превосходит  $2 \cdot 10^5$ .

В следующих  $m$  строках содержатся запросы.

Описание запроса начинается с целого числа  $t$  — типа запроса ( $1 \leq t \leq 2$ ).

Если  $t = 1$ , то далее в этой строке идут целое число  $i$  ( $1 \leq i \leq n$ ) и две строчные латинские буквы  $c_1$  и  $c_2$ ,  $c_1 \neq c_2$ , задающие, что Крош заменяет в  $i$ -й строке все буквы  $c_1$  буквами  $c_2$ .

Если  $t = 2$ , то далее в этой строке следуют два целых числа  $i$  и  $j$  числа и строчная латинская буква  $c$  ( $1 \leq i \leq j \leq \ell$ ), где  $\ell$  — суммарная длина всех строк (обратите внимание, что параметр  $\ell$  не задан во входной файле). Запрос обозначает, что требуется посчитать, сколько раз символ  $c$  встречается в строке, полученной склейкой всех строк, на позициях между  $i$ -й и  $j$ -й включительно.

### Формат выходных данных

Для каждого запроса второго типа выведите в отдельной строке одно целое число — ответ на него.

### Пример

стандартный ввод	стандартный вывод
5 5	
ab	
ac	
bbb	
de	
aba	
1 1 a b	
1 2 c a	
2 1 12 a	
1 4 e b	
2 5 10 b	
	4
	4

## Задача G. Ёлочки

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

Стилизованные ёлочки к Новому Году часто делают из заготовок в виде правильных треугольников, размеченных треугольной сеткой.

Треугольная сетка со стороной  $n$  получается разбиением правильного треугольника со стороной  $n$  на правильные треугольники со стороной 1 с помощью прямых, параллельных сторонам исходного треугольника.

Узлами сетки называются вершины правильных треугольников со стороной 1, используемых в разбиении.

Напишите программу, которая по заданным  $n$ ,  $a$  и  $b$  определяет количество правильных треугольников (не обязательно со сторонами, параллельными сторонам исходного треугольника) с вершинами в узлах сетки со стороной  $n$ , длины сторон которых заключены между значениями  $a$  и  $b$ .

### Формат входных данных

Входные данные состоят из одной строки, содержащей три целых числа  $n$ ,  $a$  и  $b$  ( $1 \leq n \leq 10^9$ ,  $1 \leq a \leq b \leq 10^6$ ).

### Формат выходных данных

Выведите остаток от деления количества правильных треугольников с вершинами в узлах сетки со стороной  $n$ , длины сторон которых заключены между значениями  $a$  и  $b$ , на 998 244 353.

### Примеры

стандартный ввод	стандартный вывод
3 2 2	3
5 1 2	50
1 2 3	0

## Задача Н. Железные дороги города К.

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	7 секунд
Ограничение по памяти:	1024 мегабайта

Город К состоит из  $n$  островов. Некоторые из этих площадей соединены железнодорожными мостами с двусторонним движением так, что между любыми двумя островами можно проехать по сети мостов. Всего в городе  $n - 1$  мостов.

Также в городе  $m$  маршрутов электричек. Каждый маршрут характеризуется начальным и конечным островом и при следовании в одном направлении не проходит по одному и тому же острову (или мосту) более одного раза.

Администрация города хочет раскрасить электрички так, чтобы все электрички одного и того же маршрута были покрашены в один цвет, и чтобы никакие два маршрута, у которых электрички покрашены в одинаковые цвета, не проходили через один и тот же остров.

Требуется найти минимальное необходимое количество цветов и построить соответствующее решение.

### Формат входных данных

Первая строка входных данных содержит одно целое число  $t$  — количество тестовых примеров ( $1 \leq t \leq 10^5$ ).

Первая строка каждого тестового примера содержит два целых числа  $n, m$  ( $1 \leq m, n \leq 5 \cdot 10^5$ ). Каждая из последующих  $n - 1$  строк содержит по два целых числа  $u_i, v_i$ , обозначающих, что  $i$ -й железнодорожный мост соединяет острова  $u_i$  и  $v_i$  ( $1 \leq u_i, v_i \leq n$ ). Каждая из последующих  $m$  строк содержит по два целых числа  $a_j$  и  $b_j$ , задающих острова, соединённые  $j$ -м маршрутом электрички ( $1 \leq a_j, b_j \leq n$ ). Гарантируется, что между любыми двумя островами можно проехать по железной дороге и что сумма  $n$  по всем тестовым примерам не превосходит  $5 \cdot 10^5$ , а также сумма  $m$  по всем тестовым примерам не превосходит  $5 \cdot 10^5$ .

### Формат выходных данных

Для каждого тестового примера выведите две строки. Первая строка должна содержать целое число  $k$  — минимальное необходимое количество цветов.

Вторая строка должна содержать  $m$  целых положительных чисел, где  $i$ -е число обозначает цвет  $c_i$ , назначенный  $i$ -му маршруту электричек так, что  $1 \leq c_i \leq k$ .

### Пример

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

## Задача I. Замена отрезков частями

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	5 секунд
Ограничение по памяти:	1024 мегабайта

Заданы  $n$  отрезков  $[a_i, b_i]$ . Затем выбирается перестановка  $p$  длины  $m$  и выполняются  $m$  операций разреза, каждая из которых описывается тройкой  $(p_i, l_i, r_i)$ , где  $p_i$  —  $i$ -й элемент перестановки  $p$ . Для каждого целого  $x \in [l_i, r_i]$  выполняется следующая операция:

- Если  $p_i \notin [a_x, b_x]$ , ничего не делать.
- Иначе отрезок  $i$  разрезается на две части  $[a_i, p_i]$  и  $[p_i, b_i]$ . В качестве нового отрезка выбирается более длинная часть; если длины равны, выбирается  $[a_i, p_i]$ .

Требуется найти левую и правую границы каждого интервала после выполнения всех операций.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $m$  ( $1 \leq n \leq 10^5$ ,  $1 \leq m \leq 10^6$ ).

Каждая из последующих  $n$  строк содержит по два целых числа  $a_i, b_i$  ( $1 \leq a_i < b_i \leq m$ ), задающих границы  $i$ -го интервала.

Следующие  $m$  строк содержат по три целых числа  $p_i, l_i, r_i$  ( $1 \leq p_i \leq m$ ,  $1 \leq l_i \leq r_i \leq n$ ), описывающие операции разреза. Гарантируется, что во всех операциях  $p_i$  попарно различны.

### Формат выходных данных

Выполните  $n$  строк. В  $i$ -й строке выведите два целых числа — левую и правую границы  $i$ -го интервала после применения всех операций.

### Пример

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