

Задача A. $\sqrt{\text{Range Minimum Query}}$

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

Компания Giggle открывает свой новый офис в Судиславле, и вы приглашены на собеседование.
Ваша задача — решить поставленную задачу.

Вам нужно создать структуру данных, которая представляет из себя массив целых чисел. Изначально массив пуст. Вам нужно поддерживать две операции:

- запрос: «? i j » — возвращает минимальный элемент между i -м и j -м, включительно;
- изменение: «+ i x » — добавить элемент x после i -го элемента списка. Если $i = 0$, то элемент добавляется в начало массива.

Конечно, эта структура должна быть достаточно хорошей.

Формат входных данных

Первая строка входного файла содержит единственное целое число n — число операций над массивом ($1 \leq n \leq 200\,000$). Следующие n строк описывают сами операции. Все операции добавления являются корректными. Все числа, хранящиеся в массиве, по модулю не превосходят 10^9 .

Формат выходных данных

Для каждой операции в отдельной строке выведите её результат.

Пример

стандартный ввод	стандартный вывод
8	4
+ 0 5	3
+ 1 3	1
+ 1 4	
? 1 2	
+ 0 2	
? 2 4	
+ 4 1	
? 3 5	

Задача В. Мощные юнги

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.8 секунд
Ограничение по памяти:	256 мегабайт

Имеется список из n юнг, для каждого из которых известен его рост a_1, a_2, \dots, a_n . Рассмотрим некоторый его подсписок a_l, a_{l+1}, \dots, a_r , где $1 \leq l \leq r \leq n$, и для каждого натурального числа s обозначим через K_s число юнг с ростом s в этом подсписке. Назовем *мощностью* подсписка сумму произведений $K_s \cdot K_s \cdot s$ по всем различным натуральным s . Так как количество различных чисел в массиве конечно, сумма содержит лишь конечное число ненулевых слагаемых.

Необходимо вычислить мощности каждого из t заданных подсписков.

Формат входных данных

Первая строка содержит два целых числа n и t ($1 \leq n, t \leq 200000$) — длина списка и количество запросов соответственно.

Вторая строка содержит n натуральных чисел a_i ($1 \leq a_i \leq 10^6$) — рост юнг.

Следующие t строк содержат по два натуральных числа l и r ($1 \leq l \leq r \leq n$) — индексы левого и правого концов соответствующего подсписка.

Формат выходных данных

Выведите t строк, где i -ая строка содержит единственное натуральное число — мощность подсписка i -го запроса.

Примеры

стандартный ввод	стандартный вывод
3 2	3
1 2 1	6
1 2	
1 3	
8 3	20
1 1 2 2 1 3 1 1	20
2 7	20
1 6	
2 7	

Задача С. Частота строки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 1.5 секунд
Ограничение по памяти: 512 мегабайт

Дана строка s . Требуется ответить на n запросов. i -й запрос состоит из целого числа k_i и строки m_i , ответом является минимальная длина строки t такой, что t является подстрокой s и строка m_i входит в t как подстрока не менее k_i раз.

Подстрокой строки называется любая последовательность подряд идущих символов в этой строке.

Гарантируется, что для любых двух запросов строки m_i из этих запросов различны.

Формат входных данных

В первой строке содержится строка s ($1 \leq |s| \leq 10^5$).

Во второй строке содержится целое число n ($1 \leq n \leq 10^5$).

В каждой из следующих n строк содержатся целое число k_i ($1 \leq k_i \leq |s|$) и непустая строка m_i — параметры запроса с номером i .

Все строки во вводе состоят только из строчных букв латинского алфавита. Суммарная длина всех строк во вводе не превосходит 10^5 . Все m_i различны.

Формат выходных данных

Для каждого запроса выведите ответ на него в отдельной строке.

Если строка m_i встречается в s менее k_i раз, выведите -1 .

Примеры

стандартный ввод	стандартный вывод
aaaaa	3
5	4
3 a	4
3 aa	-1
2 aaa	5
3 aaaa	
1 aaaaa	
abbb	-1
7	2
4 b	-1
1 ab	3
3 bb	-1
1 abb	1
2 bbb	-1
1 a	
2 abbb	

Задача D. Треугольники

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2.5 секунд
Ограничение по памяти: 256 мегабайт

Дан граф. Найдите количество циклов длины 3.

Формат входных данных

Первая строка содержит два целых числа n и m ($1 \leq n, m \leq 3 \cdot 10^5$) — количество вершин и рёбер, соответственно.

Каждая из следующих m строк содержит по два целых числа от 1 до n — вершины, которые соединяет соответствующее ребро.

Гарантируется, что в графе нет петель и кратных рёбер.

Формат выходных данных

Выведите одно число — ответ.

Пример

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

Задача Е. Нестабильность сети

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

Вася — системный администратор в большой компании под названием Глюкософт. В сети компании n компьютеров, и какие-то пары компьютеров напрямую соединены сетевыми кабелями, всего таких соединений ровно m , при чем никакой кабель не соединяет компьютер с самим собой, и между любой парой компьютеров не больше одного кабеля.

На каждом компьютере в Глюкософе установлена специальная программа, поддерживаемая разработчиками Глюкософта, под названием СетеБаг. Новые версии этой программы выходят почти каждый день, но система обновлений в компании совершенно непостижима. Так, обновление версии программы происходит не синхронно для всех компьютеров. Это происходит следующим образом — выбирается один компьютер, и на него устанавливается какая-то версия СетеБага. Поскольку система обновлений настроена очень странным образом, вполне возможно, что на компьютер установят не последнюю версию СетеБага, а какую-то другую, например, ту что уже установлена на нем, или даже какую-то более раннюю.

После нескольких месяцев работы, Вася обнаружил, что наиболее частая причина нестабильной работы сети в несовместимости версий СетеБага двух компьютеров в Глюкософе. Если два компьютера, напрямую соединенных кабелем, имеют установленный Сетебаг разных версий, то соединение между этими компьютерами *нестабильно*: попытка передачи данных между ними может вызвать отказ системы. Чем больше нестабильных соединений между компьютерами, тем больше вероятность отказа системы. С другой стороны, обмен данными между компьютерами с одинаковой версией Сетебага обычно не приводит ни к каким ошибкам.

Задача Васи — предотвратить и исправить ошибки в сети Глюкософта. Сделать он это может только если он находится на рабочем месте целый день. Вася хочет быть на работе, когда вероятность отказа системы достаточно высока. Он хочет распланировать свое расписание на следующий год, но оценка вероятностей отказа системы — довольно сложная задача.

Помогите Васе найти число нестабильных соединений между комьютерами после каждого обновления версий Сетебага.

Формат входных данных

В первой строке даны два целых числа n, m — количество компьютеров и соединений между ними ($1 \leq n, m \leq 10^5$).

Во второй строке даны n целых чисел v_1, v_2, \dots, v_n — версии Сетебага, изначально установленные на компьютеры Глюкософта.

В следующих m строках даны пары целых чисел a_i, b_i — номера компьютеров, соединенных i -м кабелем ($1 \leq a_i, b_i \leq n, a_i \neq b_i$). Гарантируется, что никакие два компьютера не соединены больше, чем одним кабелем.

В следующей строке дано целое число q — количество запланированных обновлений версий Сетебага ($1 \leq q \leq 10^5$).

В следующих q строках даны пары целых чисел c_i, v'_i — номер компьютера, на котором обновляется версия Сетебага, и новая версия Сетебага, которая установится на компьютер ($1 \leq c_i \leq n, 1 \leq v'_i \leq 10^5$). Все обновления даны в хронологическом порядке, и никакие два обновления не происходят одновременно.

Формат выходных данных

Выполните для каждого запроса изменения одно целое число — количество нестабильных соединений сразу после очередного обновления версии Сетебага.

Примеры

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

Задача F. Варенье

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

Малыш и Карлсон решили пойти на прогулку. Они знают, что прогулка будет совсем скучной, если перед ней не опустошить несколько банок варенья.

Малыш достал из кладовки n банок варенья и выставил их в ряд. В банке номер i содержится ровно a_i грамм варенья. Карлсон немного подумал и решил, что в некоторых банках недостаточно варенья, и что в банке номер i должно быть хотя бы b_i грамм варенья.

Выходить из этой ситуации Карлсон хочет в t этапов. На каждом этапе он выбирает числа l, r, x, y а затем выполняет следующие операции: в банку номер l он добавляет x грамм варенья, в банку номер $l + 1$ добавляет $x + y$ грамм варенья, в банку номер $l + 2 - x + 2 \cdot y$, и так далее. В банку номер r наш герой добавит $x + y \cdot (r - l)$ грамм варенья.

Малышу хочется определить для каждой банки i наименьший номер операции, после которой в ней станет хотя бы b_i грамм варенья. Помогите Малышу: найдите соответствующее число для каждой банки.

Формат входных данных

В первой строке входного файла задано одно число n ($1 \leq n \leq 10^5$) — количество банок. Во второй строке заданы n чисел a_i ($0 \leq a_i \leq 2 \cdot 10^9$) — изначальное количество варенья в банке номер i . В третьей строке заданы n чисел b_i ($0 \leq b_i \leq 2 \cdot 10^9$) — минимальное количество варенья, которое должно быть в банке номер i .

В четвертой строке задано t ($0 \leq t \leq 10^5$) — число этапов добавления варенья в банки, которые выполнит Карлсон. В следующих t строках описаны сами этапы в хронологическом порядке. Каждый этап задан четырьмя числами l, r, x, y ($1 \leq l \leq r \leq n, 0 \leq x, y \leq 10^5$).

Формат выходных данных

Выведите n чисел в одной строке, разделенные пробелом. Число номер i должно быть равно 0, если в банке номер i изначально было достаточно варенья, номеру этапа, после которого в ней станет хотя бы b_i варенья, или -1 , если даже после выполнения всех этапов, в этой банке будет недостаточно варенья. Этапы нумеруются с единицы.

Пример

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

Задача G. Лунки

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 0.25 секунд
Ограничение по памяти: 256 мегабайт

Маленький Петя очень любит играть. Больше всего он любит играть в игру «Лунки». Это игра для одного игрока со следующими правилами:

Есть N лунок, расположенных в ряд, пронумерованных слева направо числами от 1 до N . У каждой лунки изначально установлена своя сила выброса (у лунки с номером i она равна a_i). Если бросить шарик в лунку i , то он тут же вылетит из нее и попадет в лунку $i + a_i$, после чего он опять вылетит и т.д.. Если же лунки с таким номером нету, то он просто вылетит за край ряда. На каждом из M ходов игрок выбирает одно из двух действий:

- Установить силу выброса лунки a равной b .
- Бросить шарик в лунку a и посчитать количество прыжков шарика, прежде чем он вылетит за край ряда, а так же записать номер лунки, после выпрыгивания из которой шарик вылетел за край.

У Пети есть некоторые проблемы с математикой, поэтому, как Вы уже догадались, именно Вам предстоит произвести все подсчеты.

Формат входных данных

Первая строка содержит два числа N и M ($1 \leq N \leq 10^5$, $1 \leq M \leq 10^5$) — количество лунок в ряду и количество ходов. Следующая строка содержит N целых положительных чисел, не превышающих N — изначальные силы выброса лунок. Следующие M строк задают ходы, сделанные Петей. Каждая строка может быть двух типов:

- 0 a b
- 1 a

Тут, первый тип означает что требуется установить силу выброса лунки a равной b , а второй означает что требуется бросить мячик в лунку с номером a . Числа a и b — целые положительные и не превышают N .

Формат выходных данных

Для каждого хода второго типа (задающего брасывание шарика) в порядке следования во входном файле выведите два числа через пробел в отдельной строке — номер последней лунки перед вылетом шарика за край и количество прыжков.

Пример

стандартный ввод	стандартный вывод
8 5	8 7
1 1 1 1 1 2 8 2	8 5
1 1	7 3
0 1 3	
1 1	
0 3 4	
1 2	

Задача Н. Сумма цифр

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

Для целых чисел b и n определим функцию $f(b, n)$ так:

- Если $n < b$, то $f(b, n) = n$
- Иначе $f(b, n) = f(b, \lfloor n/b \rfloor) + (n \bmod b)$

Здесь $\lfloor n/b \rfloor$ означает максимальное целое число не превышающее n/b , а $n \bmod b$ означает остаток от деления n на b .

Менее формально $f(n, b)$ — сумма цифр числа n в записи в системе с основанием b .

Так например:

- $f(10, 87654) = 8 + 7 + 6 + 5 + 4 = 30$
- $f(100, 87654) = 8 + 76 + 54 = 138$

Вам даны целые числа n и s . Найдите минимальное такое $b \geq 2$, такое что $f(b, n) = s$. Если такого b не существует, выведите -1 .

Формат входных данных

Первая строка входных данных содержит целое число n ($1 \leq n \leq 10^{11}$) Вторая строка входных данных содержит целое число s ($1 \leq s \leq 10^{11}$)

Формат выходных данных

В единственной строке выходных данных выведите ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
1 31415926535	-1
87654 30	10
87654 138	100
87654 45678	-1
31415926535 1	31415926535

Задача I. Запросы композиции перестановок

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

Вам дан массив a_1, \dots, a_n , состоящий из перестановок длины m .

Мы можем определить операцию $+$ для двух перестановок x и y длины m как такую перестановку $z = x + y$, что $z_i = y_{x_i}$ для всех $1 \leq i \leq m$. Заметьте, что порядок сложения важен.

Вам даны q запросов, каждый запрос задается двумя числами $1 \leq l \leq r \leq n$. Рассмотрим перестановку b длины m , такую что $b = ((\dots((a_l + a_{l+1}) + a_{l+2}) + \dots) + a_r)$. Тогда ответом на запрос будет являться сумма $\sum_{i=1}^m i \cdot b_i$. Реализуйте программу, быстро отвечающую на эти запросы.

Формат входных данных

Первая строка содержит единственное целое число t равное количеству тестовых случаев ($1 \leq t \leq 1000$). Далее следует описание t тестовых случаев, каждое в следующем формате:

Первая строка каждого описания содержит два целых числа n, m ($1 \leq n, m \leq 10^5$ и $1 \leq n \cdot m \leq 2 \cdot 10^5$). Следующие n строк содержат по m различных целых чисел $a_{i1}, a_{i2}, \dots, a_{iM}$, разделенных пробелами ($1 \leq a_{ij} \leq m$). Следующая строка содержит единственное целое число q ($1 \leq q \leq 2 \cdot 10^5$). Следующие q строк содержат по два целых числа l, r , разделенных пробелами ($1 \leq l \leq r \leq n$).

Гарантируется, что сумма $n \cdot m$ и сумма q по всем тестовым случаям не превосходит $2 \cdot 10^5$.

Формат выходных данных

Выведите ответы на запросы в том порядке, к которому они заданы во входных данных.

Пример

стандартный ввод	стандартный вывод
1	10
4 3	11
3 2 1	11
1 3 2	14
1 2 3	11
2 3 1	
5	
1 1	
1 4	
3 4	
3 3	
1 3	

Задача J. Задача

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 6 секунд
Ограничение по памяти: 256 мегабайт

Даны две последовательности длины N : A_1, A_2, \dots, A_N и B_1, B_2, \dots, B_N . Требуется написать программу, выполняющую следующий тип запроса:

- $i \ j \ k$: Для всех p, q таких, что $i \leq p, q \leq j$, найдите количество пар (p, q) , для которых $A_p \times B_q \leq k$.

Формат входных данных

Первая строка содержит одно целое число N ($1 \leq N \leq 100,000$).

Вторая строка содержит последовательность A_1, A_2, \dots, A_N ($1 \leq A_i \leq 100,000$).

Третья строка содержит последовательность B_1, B_2, \dots, B_N ($1 \leq B_i \leq 100,000$).

Четвертая строка содержит количество запросов M ($1 \leq M \leq 100,000$).

Далее следуют M строк, каждая из которых содержит три целых числа i, j, k ($1 \leq i \leq j \leq N$, $1 \leq k \leq 100,000$), описывающих соответствующий запрос.

Формат выходных данных

Для каждого запроса выведите ответ на отдельной строке — количество пар (p, q) , удовлетворяющих условиям запроса.

Пример

стандартный ввод	стандартный вывод
5 1 1 1 1 1 1 1 1 1 1 1 1 5 1	25

Задача К. Турбо-кузнецик

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

Один всем известный кузнецик как всегда прыгает по кочкам и собирает монетки. Всего кочек 30001 штука, они расположены в ряд и пронумерованы от 0 до 30000. Всего на этих кочках n монет, при чем i -я из них расположена на кочке номер p_i .

Кузнецик только что прибыл на 0-ю кочку и дальше будет прыгать по следующим правилам:

Сначала он прыгнет с кочки номер 0 на кочку номер d .

Затем он продолжит прыжки следующим образом: Пусть длина предыдущего прыжка равняется l , т. е., если его предыдущий прыжок был с кочки $prev$ на кочку cur , то $l = cur - prev$. Следующим шагом он совершил прыжок вперёд длины $l - 1$, l или $l + 1$. Таким образом, он прыгнет на кочку $(cur + l - 1)$, $(cur + l)$ или $(cur + l + 1)$ (конечно, в случае, если соответствующей кочки не существует, прыгнуть нельзя). Длины прыжков должны быть положительными, то есть, нельзя совершить прыжок длины 0, если сейчас $l = 1$. Если корректного пункта назначения нет, то прыжки прекращаются.

Кузнецик собирает монетки на кочках, которые он посещает в процессе. Найдите максимальное количество монеток, которые он может собрать.

Формат входных данных

В первой следуют два целых числа через пробел, n и d ($1 \leq n, d \leq 30000$), обозначающих количество монеток на кочка и длину первого прыжка кузнечика, соответственно.

В следующих n строках следуют расположения монеток. В i -й из них ($1 \leq i \leq n$) записано целое число p_i ($d \leq p_1 \leq p_2 \leq \dots \leq p_n \leq 30000$), обозначающее номер острова, содержащего i -ю монетку.

Формат выходных данных

Выведите единственное число – максимальное число монеток, которые может собрать кузнецик

Примеры

стандартный ввод	стандартный вывод
4 10 10 21 27 27	3
8 8 9 19 28 36 45 55 66 78	6
13 7 8 8 9 16 17 17 18 21 23 24 24 26 30	4

Задача L. MEX на пути

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

Дано дерево, на каждом ребре которого написано неотрицательное целое число. Вам необходимо ответить на несколько запросов вида «для данных вершин u , v назовите наименьшее неотрицательное целое число, которое **не** встречается среди чисел, написанных на ребрах на пути от u до v ».

Формат входных данных

Первая строка входных данных содержит два числа n и q ($2 \leq n \leq 10^5$, $1 \leq q \leq 10^5$), количество вершин и количество запросов.

Следующие $n - 1$ строк содержат по три числа u_i , v_i , x_i ($1 \leq u_i, v_i \leq n$, $u_i \neq v_i$, $0 \leq x_i \leq 10^9$), которые описывают ребро дерева (u_i, v_i) , на котором написано число x_i .

Следующие q строк содержат по паре чисел a_j , b_j ($1 \leq a_j, b_j \leq n$), которая обозначает запрос на пути от a_j до b_j .

Формат выходных данных

Для каждого запроса выведите единственное число — минимальное число, которое не встречается на пути.

Пример

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

Задача M. Hubble Space Telescope's 35th anniversary

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

В далеком космосе находятся N орбитальных станций, пронумерованных от 1 до N в порядке убывания их высот над планетой. У каждой станции своя уникальная высота. Между станциями проложено M односторонних космических трансферов. i -й трансфер ($1 \leq i \leq M$) ведет от станции S_i к станции E_i , строго с более высокой орбиты на более низкую. Против направления трансферов двигаться невозможно.

В честь 35-летия телескопа «Хаббл» команда ученых организует Q праздничных космических встреч. Каждый ученый изначально закреплен за одной из станций (за каждой из станций закреплен ровно 1 ученый).

Для каждой j -й встречи ($1 \leq j \leq Q$) известно, что Y_j ученых слишком заняты важными экспериментами и не смогут прибыть. Встреча состоится на станции T_j . Ученый может попасть на встречу только в том случае, если существует путь от его станции до T_j с использованием трансферов в разрешенном направлении.

Ученые — настоящие романтики космоса: если у них есть несколько маршрутов на выбор, они выберут путь с максимальным числом трансферов, чтобы вдоволь насладиться космическими пейзажами.

Ваша задача — для каждой встречи определить, сколько трансферов использует самый длинный путь среди всех ученых, которые прибудут на встречу. Если ни один ученый не сможет прибыть на встречу, выведите -1 .

Формат входных данных

Первая строка содержит три целых числа, разделенных пробелами: N ($1 \leq N \leq 10^5$), M ($0 \leq M \leq 2 \cdot 10^5$), Q ($1 \leq Q \leq 10^5$) — количество станций, трансферов и встреч.

Следующие M строк содержат по два целых числа S_i и E_i , описывающих существующий трансфер от станции S_i к станции E_i . ($1 \leq S_i < E_i \leq N$); $(S_i, E_i) \neq (S_j, E_j) \forall 1 \leq i < j \leq M$.

Далее идут Q строк, каждая описывает одну встречу: сначала два числа T_j и Y_j , затем Y_j целых чисел $C_{j,1}, C_{j,2}, \dots, C_{j,Y_j}$ — номера станций, где работают занятые ученые.

$$1 \leq T_j, Y_j \leq N$$

$$1 \leq C_{j,1} < C_{j,2} < \dots < C_{j,Y_j} \leq N$$

$$Y_1 + Y_2 + \dots + Y_Q \leq 10^5.$$

Формат выходных данных

Выполните Q строк. В j -й строке ($1 \leq j \leq Q$) выведите одно число — максимальное количество трансферов, которое использует ученый с самым длинным маршрутом, прибывший на j -ю встречу. Если никто не сможет прибыть, выведите -1 .

Пример

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

Замечание

Пояснения к первому примеру:

На первую встречу смогут прийти ученые со станций 2, 3, 4. Их самый длинный маршрут до станции 4 занимает всего 1 трансфер. Ответ 1.

На вторую встречу смогут прийти ученые со станций 1, 4, 5. Со станции 4 можно только одним трансфером добраться до 5, а со станции 1 самый длинный путь займет 3 трансфера: $1 \rightarrow 3 \rightarrow 4 \rightarrow 5$.

На третью встречу на станции 2 сможет прийти только ученый со станции 2, поэтому ответ 0. Ответ -1 был бы, если ученый со станции, на которой проводится встреча, тоже не смог бы прийти.

Задача N. Странная сумма

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	256 мегабайт

В декабре студенту Лёше нужно готовиться к экзаменам. Поэтому он решил за несколько дней повторить все курсы.

Сегодня Лёша решил сесть за подготовку к экзамену по физике. План на сегодня — выучить последовательность a из n целочисленных физических констант. К сожалению, поскольку в физике Лёша совсем ничего не понимает, он постоянно отвлекается на какие-то странные занятия. Например, он фиксирует некоторый подотрезок последовательности a_l, a_{l+1}, \dots, a_r , и считает для него странную сумму следующим образом:

- Лёша перебирает все пары x, y , такие что $l \leq x \leq y \leq r$.
- Для каждой пары x, y Лёша добавляет в странную сумму расстояние Хэмминга между отрезками последовательности a_x, a_{x+1}, \dots, a_y и $a_l, a_{l+1}, \dots, a_{l+(y-x)}$.

Расстоянием Хэмминга между последовательностями p_1, p_2, \dots, p_k и q_1, q_2, \dots, q_k называется число индексов i , таких что $p_i \neq q_i$.

Ваша задача — помочь Лёше поскорее посчитать странные суммы для всех интересующих его отрезков $[l, r]$, чтобы он поскорее вернулся к учёбе.

Формат входных данных

В первой строке даны два числа n и q ($1 \leq n, q \leq 2 \cdot 10^5$) — длина последовательности и количество отрезков, для которых нужно посчитать странную сумму.

В следующей строке записаны числа a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^6$) — имеющаяся у Лёши последовательность.

После этого следует q строк запросов, в каждой строке находятся числа l_i и r_i ($1 \leq l_i \leq r_i \leq n$), означающие, что для отрезка с l_i -го по r_i -й символ включительно Лёше нужно посчитать странную сумму.

Формат выходных данных

В i -й строке выведите число, равное странной сумме на отрезке из запроса под номером i .

Примеры

стандартный ввод	стандартный вывод
4 4	0
1 2 1 3	1
1 1	4
2 3	8
2 4	
1 4	
7 5	10
2 1 5 6 6 2 3	7
1 4	0
4 7	4
4 4	3
1 3	
3 5	

Замечание

Рассмотрим последний запрос в первом примере. В нём последовательность Лёши равна $[1, 2, 1, 3]$, а также выбран подотрезок с 1 до 4.

Обозначим за h расстояние Хэмминга. Тогда ответом является сумма следующих величин:

- $h([1, 2, 1, 3], [1, 2, 1, 3]) = 0$
- $h([1, 2, 1], [1, 2, 1]) = 0$
- $h([2, 1, 3], [1, 2, 1]) = 3$
- $h([1, 2], [1, 2]) = 0$
- $h([2, 1], [1, 2]) = 2$
- $h([1, 3], [1, 2]) = 1$
- $h([1], [1]) = 0$
- $h([2], [1]) = 1$
- $h([1], [1]) = 0$
- $h([3], [1]) = 1$

Итоговая сумма равна 8.