

Задача А. Групповой проект

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

Всего студентов по направлению «Мировая культура» — n человек. Преподаватель дал задание — групповой проект. Для выполнения этого задания студенты должны разбиться на группы численностью от a до b человек. Скажите, можно ли разбить всех студентов на группы для выполнения проекта или преподаватель что-то перепутал.

Формат входных данных

В первой строке вводится число t ($1 \leq t \leq 100$) — количество тестовых случаев. Далее для каждого тестового случая вводится 3 целых числа n , a и b ($1 \leq n \leq 10^9$, $1 \leq a \leq b \leq n$) — общее число студентов и ограничение на число студентов в одной группе.

Формат выходных данных

Для каждого тестового случая выведите YES, если можно разбить студентов на группы и NO, если нельзя.

Пример

стандартный ввод	стандартный вывод
4	YES
10 2 3	NO
11 7 8	YES
28 4 6	YES
3 1 2	

Задача В. Изменения в строке

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

Даны две строки s и f . За один ход можно взять две произвольные буквы из алфавита и все вхождения в строку s первой буквы заменить на вторую букву, и наоборот, все вхождения второй буквы заменить на первую. Например в строке abc при выборе a и c получится строка cba , а при выборе b и z получится azc . Скажите, можно ли такими преобразованиями получить из строки s строку f .

Формат входных данных

В первой строке вводится число t ($1 \leq t \leq 100$) — число тестовых случаев. Далее для каждого тестового случая вводятся две строки равной длины, состоящие из строчных латинских букв — s и f соответственно.

Гарантируется, что суммарная длина всех строк s и f не превосходит 10^5 .

Формат выходных данных

Для каждого тестового случая выведите YES, если возможно данными преобразованиями получить из строки s строку f и NO, если нельзя.

Пример

стандартный ввод	стандартный вывод
4	YES
abcd	YES
efgh	YES
abc	NO
azc	
sssss	
fffff	
xx	
yz	

Задача С. Сокращение битов

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

Дано n k -битных чисел. За один ход можно взять два числа, и если их t -й бит равен единице, его можно превратить в ноль (в обоих числах). Определите, можно ли сделать все числа одинаковыми.

Формат входных данных

В первой строке вводится t ($1 \leq t \leq 100$) — число тестовых случаев. Далее идет описание тестовых случаев.

В первой строке для каждого тестового случая вводятся два целых числа n и k ($1 \leq n \leq 10000$, $1 \leq k \leq 10$). Далее следуют n строк — значения чисел в двоичной системе счисления, каждое длины k , возможно с ведущими нулями.

Гарантируется, что сумма n по всем тестовым случаям не превосходит 10000.

Формат выходных данных

Для каждого тестового случая выведите YES, если можно сделать все числа одинаковыми, и NO, если нельзя.

Пример

стандартный ввод	стандартный вывод
3	YES
3 3	YES
110	NO
101	
011	
2 2	
01	
01	
2 4	
0010	
0100	

Задача D. Красивые пары

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

Вася считает пару положительных целых чисел x и y красивой, если:

- $\text{НОД}(x, y) = g$
- $\text{НОК}(x, y) = l$

Ему стало интересно, сколько существует красивых пар (x, y) , помогите ему ответить на этот вопрос.

Формат входных данных

В единственной строке через пробел записаны 2 целых числа g и l ($1 \leq g, l \leq 10^9$)

Формат выходных данных

Выведите единственное число – количество красивых пар.

Примеры

стандартный ввод	стандартный вывод
1 10	4
3 15	2

Замечание

В первом тесте красивыми являются пары $(1, 10)$, $(2, 5)$, $(5, 2)$, $(10, 1)$

Во втором тесте – $(3, 15)$, $(15, 3)$

Задача Е. Ни больше ни меньше

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

Дан массив целых положительных чисел a длины n . Разбейте его на **минимально возможное** количество отрезков, чтобы каждое число было не меньше длины отрезка которому оно принадлежит. Длиной отрезка считается количество чисел в нем.

Разбиение массива на отрезки считается корректным, если каждый элемент принадлежит ровно одному отрезку.

Формат входных данных

Первая строка содержит одно целое число t ($1 \leq t \leq 1000$) — количество наборов тестовых данных. Затем следуют t наборов тестовых данных.

Первая строка набора тестовых данных содержит одно целое число n ($1 \leq n \leq 10^5$) — длину массива.

Следующая строка содержит n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$) — массив a .

Гарантируется, что сумма n по всем наборам тестовых данных не превосходит $2 \cdot 10^5$.

Формат выходных данных

Для каждого набора тестовых данных в первой строке выведите число k — количество отрезков в вашем разбиении.

Затем в следующей строке выведите k чисел $len_1, len_2, \dots, len_k$ ($1 \leq len_i \leq n$, $\sum_{i=1}^k len_i = n$) — длины отрезков в порядке слева направо.

Пример

стандартный ввод	стандартный вывод
3	3
5	1 2 2
1 3 3 3 2	3
16	1 6 9
1 9 8 7 6 7 8 9 9 9 9 9 9 9 9	3
7	2 3 2
7 2 3 4 3 2 7	

Замечание

Ответы в примере соответствуют разбиениям:

$\{[1], [3, 3], [3, 2]\}$

$\{[1], [9, 8, 7, 6, 7, 8], [9, 9, 9, 9, 9, 9, 9, 9]\}$

$\{[7, 2], [3, 4, 3], [2, 7]\}$

В первом наборе тестовых данных набор длин $\{1, 3, 1\}$, соответствующий разбиению $\{[1], [3, 3, 3], [2]\}$, также был бы корректным.

Задача F. Очередная шахматная задача

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

На шахматной доске $n \times n$ находятся m ферзей (ферзь бьет клетки на той же горизонтали, вертикали, или диагонали до ближайшей занятой). Определите, сколько пар ферзей бьют друг друга.

Формат входных данных

В первой строке находятся два целых числа n и m ($1 \leq n \leq 10^9$, $0 \leq m \leq 2 \cdot 10^5$), в каждой из следующих m строк находится пара целых чисел x, y ($1 \leq x, y \leq n$) — координаты очередного ферзя. Гарантируется, что все ферзи стоят в различных клетках поля.

Формат выходных данных

Выведите одно число — количество пар ферзей, которые бьют друг друга.

Примеры

стандартный ввод	стандартный вывод
3 3 1 1 1 2 1 3	2
6 9 1 1 1 3 1 5 3 1 3 3 3 5 5 3 5 5 6 1	18

Задача G. Поймай число, если сможешь

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

Финес и Ферб играют в игру. Они стартуют с некоторым счетом S . Каждым ходом один из них выбирает число из набора m_1, m_2, \dots, m_n и прибавляет или вычитает его из текущего счета на свой выбор. Одно и то же число можно выбирать сколько угодно раз. Ходы чередуются, первым ходит Финес.

Есть k ключевых чисел T_1, T_2, \dots, T_k . Если в какой-то момент счет становится равным любому числу из набора T , то выигрывает Ферб. Если есть стратегия, при которой такого не произойдет, и игра будет бесконечной, то выиграет Финес.

Определите кто выиграет в этой игре.

Формат входных данных

Первая строка содержит одно целое число t ($1 \leq t \leq 1000$) — количество наборов тестовых данных. Затем следуют t наборов тестовых данных.

Первая строка набора тестовых данных содержит три целых числа n, k, S ($1 \leq n \leq 10^3, 1 \leq k \leq 10^6, -10^9 \leq S \leq 10^9$) — размер набора m , размер набора T и стартовый счет.

Вторая строка содержит n целых чисел m_1, m_2, \dots, m_n ($0 \leq m_i \leq 10^9$) — набор m .

Третья строка содержит k целых чисел T_1, T_2, \dots, T_k ($-10^9 \leq T_i \leq 10^9$) — ключевые числа T .

Гарантируется, что сумма n по всем наборам тестовых данных не превосходит $2 \cdot 10^3$.

Гарантируется, что сумма k по всем наборам тестовых данных не превосходит $2 \cdot 10^6$.

Формат выходных данных

Для каждого набора тестовых данных в единственной строке выведите имя победителя: «Phineas» или «Ferb».

Пример

стандартный ввод	стандартный вывод
3	Ferb
6 1 -42	Phineas
0 6 9 42 239 999999999	Ferb
-42	
2 8 0	
1 3	
-69 -9 -6 4 8 12 20 42	
2 4 10	
1 3	
16 12 8 4	

Замечание

В первом тестовом случае счет изначально является ключевым числом, поэтому выигрывает Ферб.

Во втором тестовом случае игра будет бесконечной.

В третьем тестовом случае Фербу достаточно повторить ход Финеса, чтобы добиться ключевого числа.

Задача Н. Ручка и поле

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

Вам дано поле с координатами $(0, 0)$ в левом нижнем углу и координатами (n, m) в правом верхнем углу. В левом нижнем углу изначально стоит ручка.

Также вам дана последовательность команд s . Каждой команде соответствует свой символ. Каждый символ означает следующее:

- L — провести ручку влево на 1;
- R — провести ручку вправо на 1;
- U — провести ручку вверх на 1;
- D — провести ручку вниз на 1;

После выполнения последовательности команд на поле появляется рисунок. Нужно посчитать, сколько на рисунке будет различных отрезков длины 1, оба конца которых имеют целые координаты.

Формат входных данных

Первая строка входных данных содержит два целых числа n, m ($1 \leq n, m \leq 100$) — координаты правого верхнего угла у поля.

Вторая строка входных данных содержит одно целое число k ($1 \leq k \leq 10^5$) — количество команд.

Третья строка входных данных содержит одну строку s — последовательность команд.

Гарантируется, что длина s равна k и все символы в s являются корректными командами. Также гарантируется, что при выполнении команд ручка не выйдет за пределы поля.

Формат выходных данных

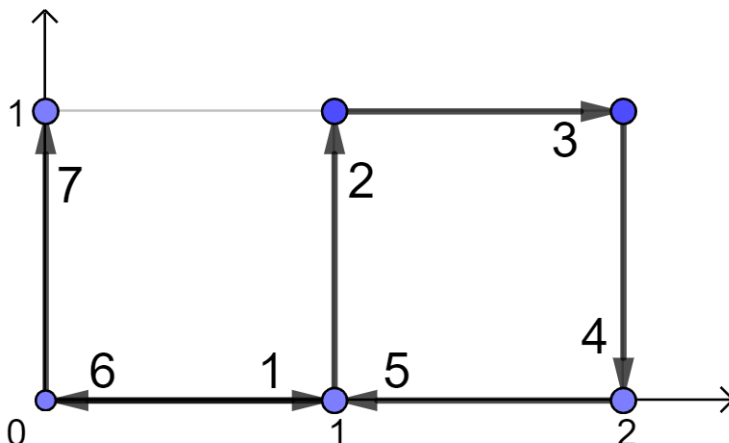
Единственная строка выходных данных должна содержать одно целое число — количество различных отрезков длины 1, оба конца которых имеют целые координаты.

Пример

стандартный ввод	стандартный вывод
2 1 7 RURDLLU	6

Замечание

Пример из условия показан на следующей картинке:



Задача I. Дизайнер Помещений

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

Егор – начинающий дизайнер помещений и сейчас занят выполнением курсовой работы. В качестве курсовой работы он выбрал обустройство коворкинга в ангаре. Сейчас ангар разбит на n комнат, расположенных подряд, длина i -й из них составляет a_i метров. Егор хочет снести некоторые из стен между соседними комнатами и сделать в каждой из получившихся комнат отдельное помещение коворкинга.

Егор считает, что среднестатистическому посетителю коворкинга будет комфортно работать в помещении, если его длина не меньше l метров и не больше r метров. Прежде чем приступить к обустройству самих помещений, Егор хочет определить степень своей творческой свободы и понять, каким количеством способов он может снести стены между соседними комнатами так, чтобы в каждом из получившихся помещений было комфортно работать. Два способа сноса стен считаются различными, если в одном из них снесена стена, которая не снесена в другом.

Так как ответ может быть большим, выведите его по модулю 998 244 353.

Формат входных данных

В первой строке вводятся три целых числа n , l и r ($1 \leq n \leq 3 \cdot 10^5$, $1 \leq l \leq r \leq 10^9$).

В следующей строке находятся n целых чисел a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$) – длины комнат, которые изначально находятся в ангаре.

Формат выходных данных

Выведите одно число – ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
5 1 9 4 6 2 4 6	3
10 1 9 4 2 4 3 1 2 3 3 5 5	117

Задача J. Модник

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

Всего в одном месяце n дней. Вы знаете, что в этом месяце будет проходить m фэшн-эвентов. i -й фэшн-эвент будет проходить со дня l_i по день r_i включительно. Во все дни i -го фэшн-эвента стоимость вашего наряда должна быть хотя бы x_i . Конечно же нельзя носить одно и то же, поэтому каждый день, когда проходит хотя бы один фэшн-эвент, вы покупаете **новый** наряд. Заметьте, что в один день может быть несколько эвентов, но для них надо купить только один наряд. Сообщите минимальное суммарное количество денег, которое потребуется, чтобы удовлетворить условия всех фэшн-эвентов.

Формат входных данных

В первой строке вводится два числа n и m ($1 \leq n \leq 10^9$, $1 \leq m \leq 10^5$). Следующие m строк описывают фэшн-эвенты. В i -й строке записаны 3 числа l_i , r_i и x_i ($1 \leq l_i \leq r_i \leq n$, $1 \leq x_i \leq 10^9$) — даты фэшн-эвента и минимальная стоимость наряда.

Формат выходных данных

Выведите минимальное суммарное количество денег, необходимое для удовлетворения условий всех фэшн-эвентов.

Примеры

стандартный ввод	стандартный вывод
10 3 1 5 5 6 10 6 3 7 7	63
3 4 1 2 2 2 3 3 2 2 1 2 2 4	9

Замечание

Заметьте, что если в данный день нет никаких фэшн-эвентов, можно не покупать новый наряд и потратить 0 денег.

Задача К. Почти бинпоиск

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

Вам дано бинарное дерево. В каждой вершине v этого дерева записано число x_v . Также вам дается q запросов. i -й запрос описывается числом t_i . Далее с этим числом t_i происходит следующий процесс:

- Вы начинаете в корне дерева — вершине 1.
- Если в вершине v , в которой вы сейчас находитесь, значение x_v больше t_i и у вершины v есть левый сын, то вы переходите в левого сына вершины v и продолжаете процесс.
- Если в вершине v , в которой вы сейчас находитесь, значение x_v меньше t_i и у вершины v есть правый сын, то вы переходите в правого сына вершины v и продолжаете процесс.
- Как только ни одно из условий не выполнено вы останавливаетесь. Восстановите, в какой вершине закончится процесс.

Формат входных данных

В первой строке вводятся 2 числа n и q ($1 \leq n, q \leq 2 \cdot 10^5$) — число вершин в дереве и число запросов. В следующей строке записаны n чисел — значения x_v ($1 \leq x_v \leq 10^9$). В следующих n строках описывается дерево. В i -й строке записаны два числа l_i и r_i ($1 \leq l_i, r_i \leq n$) — левый и правый сыновья вершины i . Если соответствующего сына нет, число будет равно 0. Гарантируется, что во вводе дано корректное бинарное дерево.

В следующих q строках описываются запросы. Для каждого запроса на новой строке вводится одно число t_i — параметр запроса.

Формат выходных данных

Для каждого запроса выведите номер вершины, в которой закончится процесс, описанный в условии.

Пример

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