

Шахматы

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

Петя любит играть в шахматы. Он знает, что самая сильная фигура в шахматах – это ферзь, потому что он ходит на все клетки на одной с ним горизонтали или вертикали и на все клетки по диагоналям. Будем считать, что все клетки имеют координаты по обеим осям – положительные целые числа, которые отсчитываются от левого нижнего угла шахматной доски, левая нижняя клетка имеет координаты $(1, 1)$. Петю заинтересовало, какое количество клеток контролирует ферзь, стоящий на клетке с координатами (x, y) на прямоугольной доске размера $m \times n$. Помогите ему в решении этой задачи.

Формат входных данных

Первая строка входных данных содержит целое число m ($1 \leq m \leq 10^9$) – размер доски по горизонтали. Вторая строка входных данных содержит целое число n ($1 \leq n \leq 10^9$) – размер доски по вертикали. Третья строка входных данных содержит целое число x ($1 \leq x \leq m$) – координата ферзя по горизонтали. Четвертая строка входных данных содержит целое число y ($1 \leq y \leq n$) – координата ферзя по вертикали.

Формат выходных данных

Программа должна вывести одно целое число – количество клеток, которое контролирует ферзь (без клетки на которой он находится).

Система оценки

Решения, правильно работающие, когда m и n не превосходят 10^4 , будут оцениваться в 70 баллов.

Пример

стандартный ввод	стандартный вывод
8 5 4 3	19

Замечание

1. Обратите внимание на то, что ответ в этой задаче может превышать возможное значение 32-битной целочисленной переменной, поэтому необходимо использовать 64-битные целочисленные типы данных (тип `long long` в языке C++, тип `int64` в Pascal, тип `long` в Java и C#).

2. В примере ферзь контролирует 7 клеток по горизонтали, 4 клетки по вертикали, по 4 клетки по обеим диагоналям, то есть всего 19 клеток.

Последовательности

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

Петя любит считать количество последовательностей, состоящих из нулей и единиц, обладающих различными свойствами. Например, сегодня он решил посчитать количество двоичных последовательностей длины N , в которых количество нулей и количество единиц не превосходят K . Помогите Пете, напишите программу, которая решает эту задачу.

Формат входных данных

Вводятся два целых числа N и K , каждое в отдельной строке ($1 \leq N, K \leq 30$).

Формат выходных данных

Выведите одно целое число — количество последовательностей.

Пример

стандартный ввод	стандартный вывод
2 1	2

Замечание

В примере последовательностей, которые отвечают требованиям Пети, всего две: 01 и 10

Проверка памяти

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

Петя решил проверить, насколько хороша память его одноклассников при помощи следующего конкурса. У него есть m карточек с числами, которые он им последовательно показывает, каждый участник конкурса для каждого нового числа должен определить, есть ли числа на предыдущих карточках, на которые делится это число, и написать «+» на листочке, если ответ положительный. Определите, чему будет равняться общее количество «+» для всех карточек.

Формат входных данных

Первая строка содержит одно целое число m ($1 \leq m \leq 10^6$) – количество показанных карточек. Вторая строка содержит m целых чисел $a_1, a_2, \dots, a_m$ ($0 < a_i \leq 10^4$) – числа на карточках.

Формат выходных данных

Выведите одно целое число — общее количество «+» для всех карточек.

Пример

стандартный ввод	стандартный вывод
5 2 3 4 6 9	3

Замечание

В тесте из условия 4 делится на 2, 6 делится на 3, 9 делится на 3, общее количество таких чисел равно трем.

Наоборот

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

Петя узнал, что произнесение букв алфавита в обратном порядке может помочь улучшить память. Он придумал свою авторскую методику, основанную на этой идее. Теперь он называет строку t , состоящую из строчных букв латинского алфавита, длины m красивой, если для любого индекса $1 \leq i < m$ t_i стоит в алфавите не раньше, чем t_{i+1} .

Петя взял n строк $s_1, s_2, \dots, s_n$, состоящих из букв от a до z и тренирует память, подсчитывая количество пар индексов (i, j) , $1 \leq i < j \leq n$, таких что строка $s_i + s_j$ является красивой, где $s_i + s_j$ – это строка, полученная склеиванием строк s_i и s_j .

Формат входных данных

Первая строка содержит одно целое число n ($1 \leq n \leq 100000$) – количество строк.

Каждая из следующих n строк содержит строку s_i . Гарантируется, что строки s_i состоят только из строчных букв от a до z .

Гарантируется, что сумма длин строк не превосходит 100000.

Формат выходных данных

Выведите число пар индексов (i, j) , $1 \leq i < j \leq n$, таких что строка $s_i + s_j$ является красивой.

Обратите внимание, что ответ в этой задаче может превышать возможное значение 32-битной целочисленной переменной, поэтому необходимо использовать 64-битные целочисленные типы данных (тип `int64` в языке Pascal, тип `long long` в C++, тип `long` в Java и C#).

Пример

стандартный ввод	стандартный вывод
4 gfe ba jdc b	1

Замечание

В примере красивой будет только 1 строка: gfeba, потому что буквы в ней стоят в порядке невозрастания номеров в алфавите. Больше таких строк получить нельзя.

Постаматы

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

В городе, где живет Петя, очень необычно организована доставка товаров. В городе есть N постаматов для хранения товаров. В i -м постамате может разместиться P_i заказов. Фура, доставляя новые товары, делает M рейсов, j -й рейс останавливается возле постамата X_j и привозит K_j заказов.

Часть заказов остается в постамате X_j . Остальные курьер доставляет в следующие постаматы в порядке увеличения номера постамата. После N -го постамата следующим является постамат с номером 1. Если в следующем постамате есть места, то часть заказов остается там. Остальные заказы курьер везет дальше.

Для каждого рейса посчитайте расходы на перевозку товаров курьером, как сумму расстояний, на которое нужно перевезти каждый заказ. Расстояние между соседними постаматами равно 1.

Первоначально все постаматы пустые и заполняются по мере выполнения рейсов.

Формат входных данных

Первая строка ввода содержит одно целое число N ($2 \leq N \leq 100000$) – количество постаматов.

Вторая строка ввода содержит N целых чисел P_i ($1 \leq P_i \leq 10^9$) – вместимость постаматов.

Третья строка ввода содержит одно целое число M ($1 \leq M \leq 100000$) – количество рейсов. Следующие M строк содержат по два целых числа – номер постамата X_j ($1 \leq X_j \leq N$), возле которого останавливается фура и количество заказов в фуре K_j ($1 \leq K_j \leq 10^9$). Гарантируется, что сумма всех K_j не превышает суммы всех P_i .

Формат выходных данных

Для каждого рейса выведите на отдельной строке расходы на перевозку заказов.

Пример

стандартный ввод	стандартный вывод
5 3 3 4 5 1 2 2 11 3 3	12 6