

Всероссийская олимпиада школьников по информатике

Вологодская область, 2024-2025 учебный год

II (муниципальный) этап

9 – 11 классы

Задача 1. Снова ремонт (100 баллов)

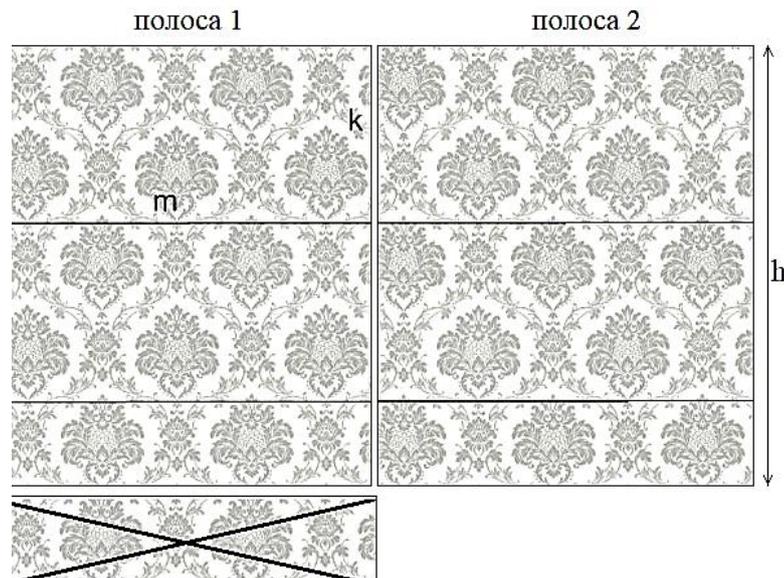
ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт
ввод / вывод: стандартный

Маша хочет оклеить обоями стену без окон и дверей длиной a и высотой h сантиметров. Маша уже выбрала красивые обои с рисунком, и теперь ей нужно сосчитать, сколько рулонов купить.

Каждый рулон имеет ширину m сантиметров и длину s сантиметров. На обоях имеется рисунок, который повторяется вдоль рулона через каждые k сантиметров.

Процесс оклейки происходит следующим образом. От рулона отрезается полоса длиной h , она будет приклеена на стену вертикально от пола до потолка. Затем отрезается следующая полоса для приклеивания рядом, и так далее. Однако, не всё так просто! Чтобы получилось красиво, нужно, чтобы рисунок на следующей полосе совместился с рисунком на предыдущей. А чтобы этого добиться, перед отрезанием следующей полосы может понадобиться вначале отрезать лишний кусок рулона (его придётся выбросить).

На рисунке показан пример. От длинного рулона отрезаны две полосы для наклеивания. При этом после отрезания первой полосы пришлось отрезать от рулона ещё кусочек (он показан перечёркнутым крест-накрест), чтобы рисунок на следующей полосе начинался точно так же, как и на предыдущей.



Заметим, что горизонтальные стыки на стенах плохо выглядят, поэтому длина приклеиваемой полосы должна всегда строго равняться высоте комнаты h . Если последняя приклеиваемая полоса окажется шире, чем оставшаяся длина комнаты, то её можно разрезать вдоль. Также заметим, что все рулоны в магазине абсолютно одинаковые.

Определите, сколько рулонов обоев надо купить Маше, чтобы ей хватило оклеить стену и при этом не взять лишние рулоны.

Входные данные. В пяти строках входных данных записаны пять целых чисел: a – длина стены, h – высота стены, k – длина повторения рисунка, m – ширина рулона обоев, s – длина рулона. Ограничения: $1 \leq a, m, s \leq 2 \cdot 10^9, h \leq s, k \leq s$.

Выходные данные. Выведите одно целое число – минимальное количество рулонов обоев, необходимое для оклейки стены.

Примеры

входные данные

9
3
3
3
20

выходные данные

1

входные данные

5
4
1
2
8

выходные данные

2

Система оценивания.

Решения, правильно работающие при дополнительном ограничении $a, m, s \leq 1000$, будут набирать не менее 50 баллов.

Задача 2. Согласование заявок (100 баллов)

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт
ввод / вывод: стандартный

Владимир Борисович работает руководителем в одной очень крупной Компании. В его служебные обязанности в числе прочего входит рассмотрение и согласование заявок, которые отправляют сотрудники через специальную информационную систему. Владимир Борисович согласовывает заявки разных типов: всего имеется n типов заявок. Он зашёл в приложение для согласования заявок и увидел, что там имеется a_1 заявок первого типа, a_2 второго и т.д.

Процесс согласования выглядит следующим образом: заявки отмечаются галочкой и нажимается кнопка «Согласовать». Кнопка работает, только когда все отмеченные заявки имеют один и тот же тип, то есть нельзя одновременно согласовать несколько заявок разных типов. Согласованные заявки исчезают из списка заявок. Нажатием на кнопку-галочку возле заявки можно не только отметить её, но и снять отметку. Есть ещё кнопка «Отметить все», которая выставляет галочки напротив всех заявок в отмеченное положение.

За какое наименьшее количество нажатий кнопок в приложении Владимир Борисович может согласовать все заявки?

Входные данные. В первой строке входных данных записано число n – количество типов заявок ($1 \leq n \leq 10^5$). В следующих n строках записано по одному числу: a_1, a_2, a_3 и так далее – количество поступивших заявок каждого типа ($1 \leq a_i \leq 10^4$ для всех i).

Выходные данные. В результате работы программы требуется вывести единственное число – минимальное возможное количество нажатий кнопок для согласования всех заявок.

Примеры

входные данные

2
3
2

выходные данные

5

входные данные

3
1
1
1

выходные данные

6

Пояснение к примеру. В первом примере выгоднее всего отметить сначала две заявки второго типа и нажать кнопку «Согласовать», потратив на это суммарно три нажатия на кнопки, после этого нажать кнопку «Отметить все» и затем кнопку «Согласовать», потратив ещё два нажатия, и таким образом согласовать все заявки, используя пять нажатий кнопок. Нетрудно проверить, что любые другие способы согласования требуют большего количества нажатий.

Система оценивания.

Решения, правильно работающие при дополнительном ограничении $n \leq 1000$, будут набирать не менее 70 баллов.

Задача 3. Треугольники (100 баллов)

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт
ввод / вывод: стандартный

Всё, что требуется в этой задаче – найти количество треугольников, у которых длины всех сторон – целые числа в диапазоне от P до Q включительно. Например, для $P=1$, $Q=2$ ответом будет 3 – это треугольники с длинами сторон $(1, 1, 1)$, $(1, 2, 2)$ и $(2, 2, 2)$.

Примечание: тройку $(1, 2, 2)$ можно было бы записать как $(2, 1, 2)$ или $(2, 2, 1)$ – это всё один и тот же треугольник.

Входные данные. Вводятся два целых числа P и Q , каждое число в отдельной строке ($1 \leq P \leq Q \leq 2 \cdot 10^5$).

Выходные данные. Выведите одно целое число – искомое количество треугольников.

Пример

входные данные

1
2

выходные данные

3

Система оценивания. Решения, правильно работающие при $Q \leq 100$, будут набирать не менее 30 баллов. Решения, правильно работающие при $Q \leq 2000$, будут набирать не менее 60 баллов.

Примечание. Обратите внимание, что ответ может превышать возможное значение 32-битной целочисленной переменной, поэтому необходимо использовать 64-битный целочисленный тип данных (тип `int64` в языке Pascal, тип `long long` в C++, тип `long` в Java и C#). В языке Python ничего дополнительно делать не требуется.

Задача 4. Маша и Глеб переезжают (100 баллов)

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт
ввод / вывод: стандартный

Маша и Глеб решили переехать в город M и задумались о приобретении там земли. Всего в продаже имеются n земельных участков, расположенных последовательно вдоль дороги. Для каждого участка известна его цена. Некоторые цены даже могут быть отрицательными – в таких случаях город сам готов заплатить покупателю, лишь бы скинуть с себя обязанности по содержанию проблемных участков.

Маша хочет выбрать себе один или несколько участков, идущих обязательно подряд, чтобы их можно было слить в один. Аналогично хочет поступить и Глеб. Кроме того, Маша и Глеб хотят стать соседями – то есть, после выполнения слияний их объединённые участки должны оказаться соседними.

Глеб очень ленив, поэтому он попросил Машу заняться подбором участков для них обоих. А Маша уже устала от лени Глеба, поэтому она решила его проучить и выбрать участки так, чтобы разница суммарных стоимостей участков Глеба и участков Маши оказалась как можно больше. Напишите программу для нахождения этой максимальной разности.

Входные данные. В первой строке входных данных записано число n ($2 \leq n \leq 10^5$). В следующих n строках записаны числа a_1, a_2, \dots, a_n – цены участков ($-10^9 \leq a_i \leq 10^9$).

Выходные данные. Выведите одно целое число – максимально возможную разницу суммарных стоимостей участков Глеба и Маши.

Примеры

входные данные

3
1
2
3

выходные данные

4

входные данные

7
1
2
-3
6
10
5
-100

выходные данные

121

Система оценивания. Решения, правильно работающие при $n \leq 10$, будут набирать не менее 15 баллов. Решения, правильно работающие при $n \leq 200$, будут набирать не менее 30 баллов. Решения, правильно работающие при $n \leq 3000$, будут набирать не менее 60 баллов.

Примечание. Обратите внимание, что ответ может превышать возможное значение 32-битной целочисленной переменной. Поэтому необходимо использовать 64-битный целочисленный тип данных (тип `int64` в языке Pascal, тип `long long` в C++, тип `long` в Java и C#). В языке Python ничего дополнительно делать не требуется.

Задача 5. Дима на рыбалке (100 баллов)

ограничение по времени на тест: 1 секунда
ограничение по памяти на тест: 256 мегабайт
ввод / вывод: стандартный

Посреди Великого озера расположился примечательный остров Фишланд. Дима очень любит рыбачить на этом острове. Чтобы улов был большим, он установил n удочек на берегу и провёл к каждой из них тропинку от центра острова. Для каждой удочки известно время, которое требуется, чтобы дойти до неё от центра острова или обратно.

Фишланд – особенный остров, поэтому через каждую минуту ровно на одну из удочек клюёт. Если в этот момент Дима находится возле этой удочки, то он вылавливает рыбу, а иначе она уплывает.

В одну из своих рыбалок Дима собрал информацию, на какую удочку клевало в каждый момент времени. Он хочет выяснить, какое максимальное количество рыб он смог бы поймать, если бы действовал оптимально. Помогите Диме это определить.

В начале рыбалки (в момент времени ноль) Дима находится в центре острова. Перемещаться между удочками он может только по тропинкам через центр. Вытаскивание рыбы из воды выполняется мгновенно.

Входные данные. Первая строка содержит одно натуральное число t ($t \leq 10^5$) – количество минут, которое длилась рыбалка.

Вторая строка содержит одно натуральное число n ($n \leq 10^5$) – количество расставленных удочек.

Следующие t строк содержат по одному числу a_i ($1 \leq a_i \leq n$) – номера удочек, на которые клевало в моменты времени 1, 2, 3 ... t .

Последние n строк содержат по одному числу d_j ($1 \leq d_j \leq t$) – за сколько минут можно добраться от центра острова до удочек с номерами 1, 2, 3 ... n .

Выходные данные. Необходимо вывести одно число – максимальное количество рыб, которое мог бы выловить Дима.

Примеры

входные данные

2
2
1
2
1
1

выходные данные

1

входные данные

6
3
3
3
2
3
2
1
1
2
3

выходные данные

2

Пояснение к примерам. В первом примере, куда бы ни пошёл Дима, удастся выловить только одну рыбу. Например, за одну минуту он может прийти до первой удочки как раз к моменту, когда на неё клюнет. Но после этого прийти до второй удочки он уже не успеет, так как на неё клюнет в момент времени 2, то есть уже через минуту, а идти до неё — две минуты.

Во втором примере Диме выгодно с самого начала пойти ко второй удочке и ловить рыбу, находясь у неё до конца рыбалки.

Система оценивания.

Подзадача 1 (до 10 баллов): $n = 1$.

Подзадача 2 (до 20 баллов): $n = 2$.

Подзадача 3 (до 20 баллов): $n, t \leq 100$.

Подзадача 4 (до 20 баллов): все $d_j = 1$.

Подзадача 5 (до 30 баллов): нет дополнительных ограничений.