

Задача А. Экзамен

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

В самом лучшем университете России есть специальный предмет, который называется «Теория Лени». Вы очень любите этот предмет и стараетесь постоянно использовать то, чему вас там научили.

Но, как и везде, на нём есть устный экзамен. Всего есть n билетов, из которых вы выучили ровно a (ваша лень не позволяет вам выучить больше).

Экзамен проходит в стандартном формате: есть стопка с билетами, каждый билет встречается в ней ровно один раз, и каждый студент случайно выбирает себе один билет. При этом, когда студент достал билет, то он забирает его себе и не возвращает обратно в стопку.

Вы знаете, что до вас отвечали уже b человек, а это значит, что стопка содержит уже на b билетов меньше. Так как вас интересует не только «Теория Лени», но и математика (и даже чуть-чуть информатика!), вы хотите узнать, какое минимальное и максимальное количество билетов из оставшихся вы можете знать.

Формат входных данных

Первая строка содержит одно целое число n ($1 \leq n \leq 10^9$) — количество билетов на экзамене.

Вторая строка содержит одно целое число a ($1 \leq a \leq n$) — количество билетов, которые вы выучили.

Третья строка содержит одно целое число b ($0 \leq b < n$) — количество людей, которые уже взяли свой билет до вас.

Формат выходных данных

Вывод вашей программы должен состоять из двух строк.

Первая строка должна содержать единственное целое число — минимальное количество билетов, которое вы можете знать из оставшихся.

Вторая строка должна содержать единственное целое число — максимальное количество билетов, которое вы можете знать из оставшихся.

Система оценки

В данной задаче 10 тестов, помимо тестов из условия, каждый из них оценивается в 10 баллов. Результаты работы ваших решений на всех тестах будут доступны сразу во время соревнования.

Решения, корректно работающие при $a = 1$, наберут не менее 20 баллов.

Решения, корректно работающие при $b = 1$, наберут не менее 20 баллов.

Примеры

стандартный ввод	стандартный вывод
6	1
4	3
3	
20	6
13	13
7	
16	8
12	12
4	

Замечание

В первом примере давайте считать, что вы знаете билеты с номерами 1, 2, 3, 4. Тогда, если люди до вас вытянули билеты с номерами 1, 2, 3, то остался только 1 билет, который вы знаете. А если люди до вас вытянули билеты с номерами 4, 5, 6, то вы знаете 3 билета из оставшихся.

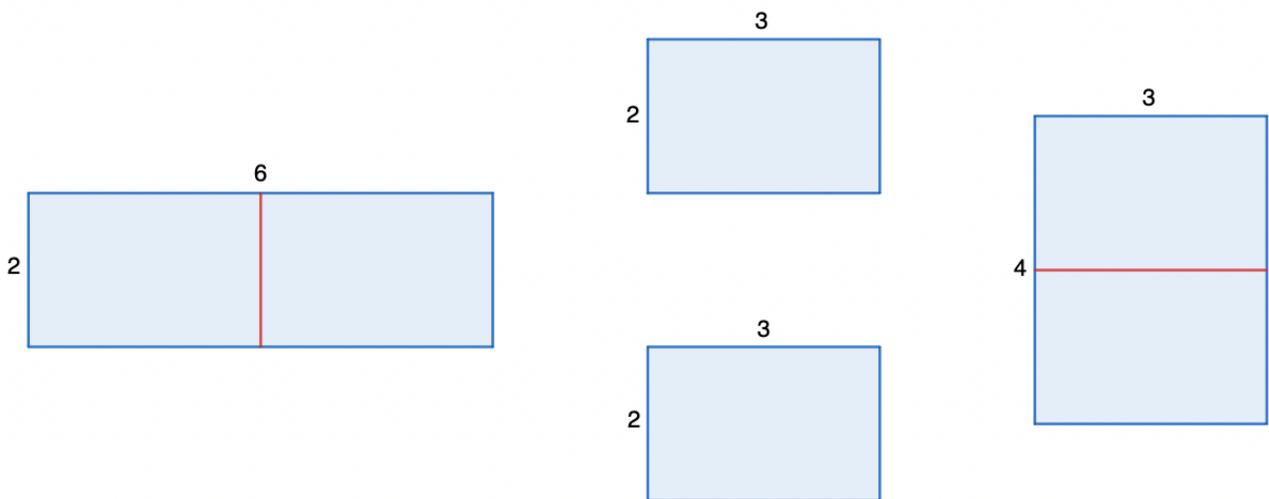
Задача В. Разрез прямоугольника

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

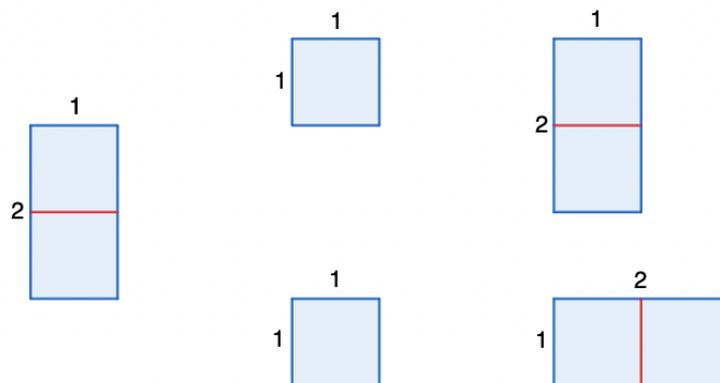
У Пети есть прямоугольник размера $a \times b$ с целыми сторонами, хотя бы одна из которых больше 1. Он пробует разрезать этот прямоугольник на два прямоугольника с целыми сторонами, сделав разрез, параллельный какой-то из сторон исходного прямоугольника. Затем Петя пытается из двух получившихся прямоугольников сложить какой-то **отличный от исходного** прямоугольник, при этом он может как угодно поворачивать и двигать эти два прямоугольника. Если у него получается это сделать, то он называет прямоугольник $a \times b$ *интересным*.

Обратите внимание, что если два прямоугольника отличаются поворотом на 90° , то они считаются **одинаковыми**. Например, прямоугольники 6×4 и 4×6 считаются одинаковыми.

Таким образом, прямоугольник 2×6 является интересным, потому что его можно разрезать на два прямоугольника 2×3 , после чего из этих двух прямоугольников сложить прямоугольник 4×3 , который отличается от прямоугольника 2×6 .



При этом прямоугольник 2×1 не является интересным, потому что его можно разрезать только на два прямоугольника 1×1 , а из них можно сложить только прямоугольники 1×2 и 2×1 , которые считаются одинаковыми с исходным.



Также у Пети есть некоторое целое число n . Он хочет узнать, сколько существует **различных** интересных прямоугольников со сторонами, которые являются целыми числами, не превосходящими n . Помогите ему это сделать.

Формат входных данных

Первая и единственная строка содержит одно целое число n ($2 \leq n \leq 2 \cdot 10^9$) — ограничение на длину сторон прямоугольника.

Формат выходных данных

Выведите одно целое число — количество различных интересных прямоугольников с длинами сторон, не превышающими n .

Обратите внимание, что ответ может быть больше, чем возможное значение 32-битной целочисленной переменной, поэтому необходимо использовать 64-битные целочисленные типы данных (тип `int64` в языке Pascal, тип `long long` в C и C++, тип `long` в Java и C#). Язык Python будет корректно работать.

Система оценки

В данной задаче 25 тестов, помимо тестов из условия, каждый из них оценивается в 4 балла. Результаты работы ваших решений на всех тестах будут доступны сразу во время соревнования.

Решения, корректно работающие при $n \leq 2000$, наберут не менее 24 баллов.

Решения, корректно работающие при $n \leq 10^6$, наберут не менее 56 баллов.

Примеры

стандартный ввод	стандартный вывод
2	1
3	2
4	6
7	16

Замечание

В первом примере только прямоугольник 2×2 является интересным: его можно разрезать на два прямоугольника 1×2 , а из них можно сложить прямоугольник 1×4 . Обратите внимание, что прямоугольник 1×1 не является интересным, потому что хотя бы одна сторона должна быть больше 1.

Во втором примере прямоугольники 2×2 и 2×3 являются интересными. Прямоугольник 2×3 можно разрезать на два прямоугольника 1×3 , а из них можно сложить прямоугольник 1×6 . Прямоугольник 3×3 не является интересным, потому что его можно разрезать только на два прямоугольника 1×3 и 2×3 , но из них можно сложить только прямоугольник 3×3 . Обратите внимание, что прямоугольники 2×3 и 3×2 считаются одинаковыми, поэтому в ответе их нужно учесть только один раз.

Задача С. Урок физкультуры

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

В известной школе прошёл урок физкультуры. Как полагается, всех построили в шеренгу и попросили рассчитаться на «первый- k -й».

Как известно, расчёт на «первый- k -й» происходит следующим образом: первые k человек имеют номера $1, 2, 3, \dots, k$, следующие $k-1$ человек имеют номера $k-1, k-2, \dots, 1$, следующие $k-1$ человек имеют номера $2, 3, \dots, k$ и т.д. Таким образом, расчёт повторяется через каждые $2k-2$ позиции. Примеры расчёта приведены в разделе «Замечание».

Мальчик Вася постоянно всё забывает. Например, он забыл позицию, которую занимал в шеренге. Но он помнит число k , описанное выше, номер, который он получил при расчёте, а также, что его позиция в шеренге была не больше n . Другими словами, если Вася стоял на позиции y в шеренге, то $y \leq n$. Помогите Васе понять, сколько есть различных позиций в ряду, где он мог стоять.

Формат входных данных

Первая строка содержит одно целое число k ($2 \leq k \leq 10^9$) — характеристика расчёта, описанная в условии.

Вторая строка содержит одно целое число x ($1 \leq x \leq k$) — номер, который Вася получил при расчёте.

Третья строка содержит одно целое число n ($x \leq n \leq 10^9$) — верхнее ограничение на позицию Васи.

Формат выходных данных

Выведите единственное целое число — количество различных позиций, которые подходят под данные ограничения.

Система оценки

В данной задаче 20 тестов, помимо тестов из условия, каждый из них оценивается в 5 баллов. Результаты работы ваших решений на всех тестах будут доступны сразу во время соревнования.

Решения, корректно работающие при 100 000, наберут не менее 60 баллов.

Примеры

стандартный ввод	стандартный вывод
2 2 10	5
3 2 10	5
5 3 10	2

Замечание

В первом примере подходят позиции равные 2, 4, 6, 8, 10.

Во втором примере подходят позиции равные 2, 4, 6, 8, 10.

В третьем примере подходят позиции равные 3 и 7.

Пример расчёта для $k = 2$, $k = 3$ и $k = 5$:

k\№	1	2	3	4	5	6	7	8	9	10
2	1	2	1	2	1	2	1	2	1	2
3	1	2	3	2	1	2	3	2	1	2
5	1	2	3	4	5	4	3	2	1	2

Задача D. Самокат

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

В городе Новый Нижнийгород открылась новая служба доставки еды с оригинальным названием «Камосат». Курьеры этой службы передвигаются на самокатах и стремятся максимально эффективно доставлять заказы клиентам.

Для того чтобы упростить задачу планирования маршрутов, был разработан алгоритм, основанный на топографии города. Город расположен вдоль реки, поэтому его можно представить одномерным массивом, где каждый элемент массива — это высота местности в соответствующей точке. Расстояние между двумя соседними точками считается равным 1.

Каждый курьер начинает свой маршрут в некоторой точке города. Он может двигаться только вправо (по увеличению номеров элементов массива) и посещать только те точки, где высота **меньше**, чем в точке старта. Курьер стремится проехать как можно дальше, учитывая данное условие. Курьер, встретив местность **не ниже**, чем точка старта, не может продолжить путь.

Помогите курьерам определить длину максимального маршрута, по которому они могут проехать, выбрав произвольную точку старта.

Формат входных данных

Первая строка содержит одно целое число n ($2 \leq n \leq 300\,000$) — количество точек в городе.

Вторая строка содержит n целых чисел h_1, h_2, \dots, h_n ($-10^9 \leq h_i \leq 10^9$) — высоты точек города.

Формат выходных данных

Выведите одно целое число — максимальное расстояние, которое может преодолеть курьер.

Система оценки

В данной задаче 50 тестов, помимо тестов из условия, каждый из них оценивается в 2 балла. Результаты работы ваших решений на всех тестах будут доступны сразу во время соревнования.

Решения, корректно работающие при монотонном массиве высот (т.е. когда для всех $1 \leq i < n$ выполняется $h_i \leq h_{i+1}$ или для всех $1 \leq i < n$ выполняется $h_i \geq h_{i+1}$), наберут не менее 14 баллов.

Решения, корректно работающие при $n \leq 2000$, наберут не менее 28 баллов.

Примеры

стандартный ввод	стандартный вывод
7 5 4 6 2 1 9 3	2
2 1 1	0
5 4 5 2 3 4	3

Замечание

В первом примере курьер может стартовать в третьей точке с высотой 6 и проехать по высотам $6 \rightarrow 2 \rightarrow 1$.

Во втором примере курьер не может никуда проехать, потому что можно посещать только те точки, где высота меньше, чем в точке старта.

В третьем примере курьер может проехать по высотам $5 \rightarrow 2 \rightarrow 3 \rightarrow 4$.

Задача Е. Подземелья Одинокой горы

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

Однажды люди, эльфы, гномы и другие жители Средиземья собрались отнять у Смога украденные у них сокровища. Во имя этой великой цели они сплотились вокруг сильного эльфа Тимофея и начали планировать свержение правителя Одинокой горы.

Армия жителей Средиземья будет состоять из нескольких отрядов. Известно, что каждая пара существ **одной расы**, которые находятся в разных отрядах, прибавляет b единиц к суммарной силе армии. Но так как Тимофею будет сложно руководить армией, состоящей из большого числа отрядов, то суммарная сила армии, состоящей из k отрядов, уменьшается на $(k - 1) \cdot X$ единиц. Обратите внимание, что армия всегда состоит **из хотя бы одного отряда**.

Известно, что в Средиземье проживают n рас, и количество существ i -й расы равно c_i . Помогите жителям Средиземья определить максимальную силу армии, которую они могут составить.

Формат входных данных

Первая строка входных данных содержит три целых числа n , b и X ($1 \leq n \leq 200\,000$, $1 \leq b \leq 10^6$, $0 \leq X \leq 10^9$) — количество рас и константы b и X , описанные выше.

Вторая строка содержит n целых чисел c_1, c_2, \dots, c_n ($1 \leq c_i \leq 200\,000$) — количество существ каждой из n рас.

Гарантируется, что $c_1 + c_2 + \dots + c_n \leq 200\,000$.

Формат выходных данных

Выведите одно целое число — максимальную силу армии, которую могут составить жители Средиземья.

Обратите внимание, что ответ может быть больше, чем возможное значение 32-битной целочисленной переменной, поэтому необходимо использовать 64-битные целочисленные типы данных (тип `int64` в языке Pascal, тип `long long` в C и C++, тип `long` в Java и C#). Язык Python будет корректно работать.

Система оценки

В данной задаче 50 тестов, помимо тестов из условия. Результаты работы ваших решений на всех тестах будут доступны сразу во время соревнования.

Решения, корректно работающие при $X = 0$, наберут не менее 16 баллов.

Решения, корректно работающие при $n = 1$, наберут не менее 10 баллов.

Решения, корректно работающие при $n \leq 2$, наберут не менее 20 баллов.

Решения, корректно работающие при $c_1 = c_2 = \dots = c_n$, наберут не менее 14 баллов.

Решения, корректно работающие при $c_1 + c_2 + \dots + c_n \leq 2000$, наберут не менее 18 баллов.

Примеры

стандартный ввод	стандартный вывод
3 1 0 1 2 3	4
3 5 10 2 5 3	40

Замечание

В первом примере жители Средиземья могут составить 3 отряда. Так как $X = 0$, то сила армии не уменьшится из-за количества отрядов. Далее жителей по отрядам можно распределить так:

- Единственного представителя первой расы можно отправить в первый отряд.

- Первого представителя второй расы можно отправить в первый отряд, второго представителя второй расы можно отправить во второй отряд. Тогда суммарная сила армии увеличится на $b = 1$.
- Первого представителя третьей расы можно отправить в первый отряд, второго представителя третьей расы можно отправить во второй отряд, третьего представителя третьей расы можно отправить в третий отряд. Тогда суммарная сила армии увеличится на $3 \cdot b = 3$, так как они образуют три пары, находящиеся в разных отрядах.

Таким образом, суммарная сила армии равна 4.