

Задача А. Count me in

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

Вам дам массив, состоящий из чисел от 1 до 10. Отсортируйте его!

Формат входных данных

В единственной строке входных данных дан исходный массив. Количество элементов в нём не превосходит 10^6 .

Формат выходных данных

В единственной строке выходных данных выведите отсортированный массив.

Пример

стандартный ввод	стандартный вывод
10 10 2 5 1	1 2 5 10 10

Задача В. Инверсии

Имя входного файла: `inverse.in`
Имя выходного файла: `inverse.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Напишите программу, которая для заданного массива $A = \langle a_1, a_2, \dots, a_n \rangle$ находит количество пар (i, j) таких, что $i < j$ и $a_i > a_j$.

Обратите внимание на то, что ответ может не влезать в `int`.

Формат входных данных

Первая строка входного файла содержит натуральное число n ($1 \leq n \leq 100\,000$) — количество элементов массива. Вторая строка содержит n попарно различных элементов массива A — целых неотрицательных чисел, не превосходящих 10^9 .

Формат выходных данных

В выходной файл выведите одно число — ответ на задачу.

Примеры

<code>inverse.in</code>	<code>inverse.out</code>
5 6 11 18 28 31	0
5 179 4 3 2 1	10

Задача С. Мега-инверсии

Имя входного файла: mega.in
Имя выходного файла: mega.out
Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Инверсией в перестановке p_1, p_2, \dots, p_N называется пара (i, j) такая, что $i < j$ и $p_i > p_j$. Назовём мега-инверсией в перестановке p_1, p_2, \dots, p_N тройку (i, j, k) такую, что $i < j < k$ и $p_i > p_j > p_k$. Напишите алгоритм для быстрого подсчёта количества мега-инверсий в перестановке.

Формат входных данных

Первая строка входного файла содержит целое число N ($1 \leq N \leq 100\,000$). Следующие N чисел описывают перестановку: p_1, p_2, \dots, p_N ($1 \leq p_i \leq N$), все p_i попарно различны. Числа разделяются переводами строк.

Формат выходных данных

Единственная строка выходного файла должна содержать одно число, равное количеству мега-инверсий в перестановке p_1, p_2, \dots, p_N .

Примеры

mega.in	mega.out
4 4 3 2 1	4
3 1 3 2	0

Задача D. Минимизируем максимум

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 3 секунды
Ограничение по памяти: 512 мегабайт

Даны n нестрого возрастающих массивов A_i и m нестрого убывающих массивов B_j . Все массивы имеют одну и ту же длину l . Далее даны q запросов вида (i, j) , ответ на запрос – такое k , что $\max(A_{ik}, B_{jk})$ минимален. Если таких k несколько, можно вернуть любое.

Формат входных данных

На первой строке числа n, m, l ($1 \leq n, m \leq 900; 1 \leq l \leq 3000$). Следующие n строк содержат описания массивов A_i . Каждый массив описывается перечислением l элементов. Элементы массива – целые числа от 0 до $10^5 - 1$. Далее число m и описание массивов B_j в таком же формате. Массивы и элементы внутри массива нумеруются с 1. На следующей строке число запросов q ($1 \leq q \leq n \cdot m$). Следующие q строк содержат пары чисел i, j ($1 \leq i \leq n, 1 \leq j \leq m$).

Формат выходных данных

Выведите q чисел от 1 до l – ответы на запросы.

Пример

стандартный ввод	стандартный вывод
4 3 5	3
1 2 3 4 5	4
1 1 1 1 1	3
0 99999 99999 99999 99999	5
0 0 0 0 99999	4
5 4 3 2 1	3
99999 99999 99999 0 0	1
99999 99999 0 0 0	2
12	2
1 1	4
1 2	4
1 3	3
2 1	
2 2	
2 3	
3 1	
3 2	
3 3	
4 1	
4 2	
4 3	

Задача Е. Дремучий лес

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

Чтобы помешать появлению СЭС в лагере, администрация ЛКШ перекопала единственную дорогу, соединяющую «Берендеевы поляны» с Судиславлем, теперь проехать по ней невозможно. Однако, трудности не остановили инспекцию, хотя для СЭС остается только одна возможность — дойти до лагеря пешком. Как известно, Судиславль находится в поле, а «Берендеевы поляны» — в лесу.

- Судиславль находится в точке с координатами $(0, 1)$.
- «Берендеевы поляны» находятся в точке с координатами $(1, 0)$.
- Граница между лесом и полем — горизонтальная прямая $y = a$, где a — некоторое число $(0 < a < 1)$.
- Скорость передвижения СЭС по полю составляет V_p , скорость передвижения по лесу — V_f . Вдоль границы можно двигаться как по лесу, так и по полю.

Администрация ЛКШ хочет узнать, сколько времени у нее осталось для подготовки к визиту СЭС. Она попросила вас выяснить, в какой точке инспекция СЭС должна войти в лес, чтобы дойти до «Берендеевых полей» как можно быстрее.

Формат входных данных

В первой строке входного файла содержатся два положительных целых числа V_p и V_f ($1 \leq V_p, V_f \leq 10^5$). Во второй строке содержится единственное вещественное число — координата по оси O_y границы между лесом и полем a ($0 < a < 1$)

Формат выходных данных

В единственной строке выходного файла выведите вещественное число с точностью не менее 6 знаков после запятой — координата по оси O_x точки, в которой инспекция СЭС должна войти в лес.

Примеры

стандартный ввод	стандартный вывод
5 3 0.4	0.7833106
5 5 0.5	0.5000000

Задача F. Поиск позиции

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

В шеренгу друг за другом стоят n человек, рост i -го из них равен a_i условных единиц. Вы тоже собираетесь встать в эту шеренгу, при чем вам хочется встать на такую позицию p , чтобы $f(p) =$ [количество людей слева от вас того же роста, что и вы] умножить на [количество людей справа от вас с ростом, не равным вашему] было максимально.

Для этого вы можете встать в начало шеренги, в её конец, или между любыми двумя соседними людьми.

К сожалению вы не можете точно вспомнить ваш рост, у вас есть только m предположений о том, каким он может быть, и для каждого из них вы хотели бы знать оптимальную позицию, на которую вам стоило бы встать.

Формат входных данных

В первой строке даны два целых числа n и m ($1 \leq m, n \leq 10^5$). Во второй строке даны n целых чисел a_i - рост людей, стоящих в шеренге ($1 \leq a_i \leq 10^5$) В третьей строке даны m целых чисел x_i - ваш предполагаемый рост ($1 \leq x_i \leq 10^5$)

Формат выходных данных

В единственной строке выведите выведите m целых чисел - значение $f(p)$ в оптимальной для данного роста позиции.

Примеры

стандартный ввод	стандартный вывод
4 3 1 1 2 3 1 2 4	4 1 0
9 3 1 1 2 2 2 2 1 1 1 1 2 1	8 12 8

Задача G. Oracle соло мид

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

Меха-Наруто играет в компьютерную игру. У его персонажа есть следующая способность: нанести вражескому герою a единиц урона, затем восполнять ему b единиц здоровья в конце каждой секунды, начиная со следующей, на протяжении ровно c секунд. В частности, если эта способность применяется в момент времени t , то здоровье врага уменьшается на a в момент времени t , а затем увеличивается на b в моменты $t + 1, t + 2, \dots, t + c$.

У способности есть время перезарядки, равное d секундам, то есть если Меха-Наруто применяет способность в момент времени t , то в следующий раз он может её применить не раньше момента $t + d$. По некоторым причинам Меха-Наруто может использовать способность только в целые моменты времени, поэтому все изменения здоровья врага также происходят в целочисленные моменты.

Эффекты от разных применений заклинания накладываются друг на друга. В частности, если вражеский герой находится под действием k заклинаний, применённых ранее и ещё не истёкших, то его здоровье увеличится на $k \cdot b$. Помимо этого все изменения, которые происходят в один и тот же момент времени, учитываются одновременно.

Теперь Меха-Наруто интересно, может ли он убить своего оппонента просто применяя свою способность так часто, как только можно (то есть каждые d секунд). Герой считается погибшим, если уровень его здоровья становится равным 0 или ниже. Предположим, что здоровье вражеского персонажа не изменяется никаким образом, кроме как от применения заклинания. Какое наибольшее количество здоровья может быть у врага, чтобы Меха-Наруто мог его убить?

Формат входных данных

Первая строка содержит единственное число t ($1 \leq t \leq 10^5$) — число тестов.

Каждый тест описывается четвёркой натуральных чисел a, b, c и d ($1 \leq a, b, c, d \leq 10^6$), записанных через пробел и означающих соответственно мгновенный урон от способности, ежесекундно восполняемый объём здоровья, время действия каждого заклинания и время перезарядки способности.

Формат выходных данных

Для каждого теста выведите на отдельной строке -1 , если способность может рано или поздно убить любого врага, каким бы большим ни был его уровень здоровья; в противном случае выведите наибольшее число здоровья, при котором оппонент будет убит.

Пример

стандартный ввод	стандартный вывод
7	1
1 1 1 1	2
2 2 2 2	1
1 2 3 4	5
4 3 2 1	534
228 21 11 3	-1
239 21 11 3	500000500000
1000000 1 1000000 1	

Замечание

В первом тесте из условия любая единица урона отменяется через секунду, поэтому Меха-Наруто не может нанести больше, чем 1 единицу урона.

В четвёртом тесте из условия герой оппонента получает:

- 4 урона (1-е применение способности) в момент 0;

- 4 урона (2-е применение способности), и 3 единицы здоровья восполняются (1-е применение) в момент 1 (всего 5 урона к начальному уровню здоровья);
- 4 урона (3-е применение способности), и 6 здоровья восполняется (1-е и 2-е применения) в момент 2 (всего 3 урона к изначальному здоровью);
- и так далее.

Можно доказать, что ни к какому моменту времени враг не получит суммарно 6 или больше урона, поэтому ответ на этот тест есть 5. Обратите внимание, как производится пересчёт здоровья: например, если бы у врага было 8 здоровья, он бы **не** умер в момент времени 1, как если бы мы сначала вычли из его здоровья 4 единицы, а затем сочли бы его мёртвым, не успев добавить 3 единицы от лечения.

В шестом тесте из условия герой со сколько угодно большим количеством здоровья рано или поздно умрёт.

В седьмом тесте из условия ответ не помещается в 32-битный целочисленный тип.

Задача N. When democracy fails

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

В одной демократической стране приближаются парламентские выборы. Выборы проходят по следующей схеме: каждый житель страны, достигший восемнадцатилетнего возраста, отдает свой голос за одну из политических партий. После этого партия, которая набрала максимальное количество голосов, считается победившей на выборах и формирует правительство. Если несколько партий набрали одинаковое максимальное количество голосов, то они должны сформировать коалиционное правительство, что обычно приводит к длительным переговорам.

Один бизнесмен решил выгодно вложить свои средства и собрался поддержать на выборах некоторые партии. В результате поддержки он планирует добиться победы одной из этих партий, которая затем сформирует правительство, которое будет действовать в его интересах. При этом возможность формирования коалиционного правительства его не устраивает, поэтому он планирует добиться строгой победы одной из партий.

Чтобы повлиять на исход выборов, бизнесмен собирается выделить деньги на агитационную работу среди жителей страны. Исследование рынка показало, что для того, чтобы один житель сменил свои политические воззрения, требуется потратить одну условную единицу. Кроме того, чтобы i -я партия в случае победы сформировала правительство, которое будет действовать в интересах бизнесмена, необходимо дать лидеру этой партии взятку в размере p_i условных единиц. При этом некоторые партии оказались идеологически устойчивыми и не согласны на сотрудничество с бизнесменом ни за какие деньги.

По результатам последних опросов известно, сколько граждан планируют проголосовать за каждую партию перед началом агитационной компании. Помогите бизнесмену выбрать, какую партию следует подкупить, и какое количество граждан придется убедить сменить свои политические воззрения, чтобы выбранная партия победила, учитывая, что бизнесмен хочет потратить на всю операцию минимальное количество денег.

Формат входных данных

В первой строке вводится целое число n – количество партий ($1 \leq n \leq 10^5$). Следующие n строк описывают партии. Каждая из этих строк содержит по два целых числа: r_i – количество жителей, которые собираются проголосовать за эту партию перед началом агитационной компании, и b_i – взятка, которую необходимо дать лидеру партии для того, чтобы сформированное ей в случае победы правительство действовало в интересах бизнесмена ($1 \leq r_i \leq 10^6$, $1 \leq b_i \leq 10^6$ или $b_i = -1$). Если партия является идеологически устойчивой, то $b_i = -1$. Гарантируется, что хотя бы одно b_i не равно -1 .

Формат выходных данных

В первой строке выведите минимальную сумму, которую придется потратить бизнесмену. Во второй строке выведите номер партии, лидеру которой следует дать взятку. В третьей строке выведите n целых чисел – количество голосов, которые будут отданы за каждую из партий после осуществления операции. Если оптимальных решений несколько, выведите любое.

Примеры

стандартный ввод	стандартный вывод
3 7 -1 2 8 1 2	6 3 3 2 5
2 239 239 238 -1	239 1 239 238