

Задача А. Калила и Димна на лесозаготовках

Имя входного файла: `lumber.in`
Имя выходного файла: `lumber.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Спонсор сегодняшней задачи — codeforces round 189. Codeforces — мечты сбываются!

Калила и Димна — два шакала. Они живут в огромных джунглях. Однажды шакалы решили устроиться на завод лесозаготовки и подработать.

Управляющий завода хочет, чтобы они отправились в джунгли и срубили n деревьев высотой $a_1, a_2, \dots, a_n$. Для этого Калила и Димна купили цепную пилу в магазине. Каждый раз, когда они используют пилу на дереве номер i , они уменьшают высоту этого дерева на единицу. Каждый раз Калила и Димна должны заправить пилу для использования. Цена заправки зависит от того, какие деревья полностью спилены (дерево считается полностью спиленным, если его высота равна 0). Если максимальный идентификатор полностью срубленного дерева равняется i (первоначально это дерево имело высоту a_i), то цена заправки пилы равняется b_i . Если ни одно дерево не срублено полностью, то заправлять пилу запрещается. Изначально пила заправлена. Известно, что для каждого $i < j$, $a_i < a_j$ и $b_i > b_j$, а также $b_n = 0$ и $a_1 = 1$.

Калила и Димна хотят полностью срубить все деревья с минимальными затратами. Они ждут Вашей помощи! Поможете?

Формат входных данных

В первой строке записано целое число n ($1 \leq n \leq 10^5$). Во второй строке записано n целых чисел $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^9$). В третьей строке записано n целых чисел $b_1, b_2, \dots, b_n$ ($0 \leq b_i \leq 10^9$).

Гарантируется, что $a_1 = 1$, $b_n = 0$, $a_1 < a_2 < \dots < a_n$ и $b_1 > b_2 > \dots > b_n$.

Формат выходных данных

В единственной строке должна быть записана минимальная стоимость вырубания всех деревьев.

Примеры

<code>lumber.in</code>	<code>lumber.out</code>
5 1 2 3 4 5 5 4 3 2 0	25
6 1 2 3 10 20 30 6 5 4 3 2 0	138

Задача В. Транспортировка кошек

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

Zxr960115 содержит большое хозяйство. Он кормит m милых кошечек и держит у себя p кормильщиков. Через ферму проходит прямая дорога, а вдоль дороги расположено n холмов, пронумерованных от 1 до n , слева направо. Расстояние от холма i до $i-1$ равняется d_i метров. Кормильщики живут на холме 1.

Однажды кошечкам захотелось порезвиться и они разбежались. Кошка i пошла к холму h_i , дошла до него в момент времени t_i , а затем стала ждать кормильщика на холме h_i . Кормильщики должны собрать всех разбежавшихся кошек. Каждый кормильщик идет прямо от холма номер 1 до холма номер n , не останавливаясь у какого-либо холма, и собирает всех кошек, **ожидающих** на каждом холме. Кормильщики двигаются со скоростью 1 в единицу времени и достаточно сильны, чтобы собрать сколько угодно кошек.

Например, пусть имеется два холма ($d_2 = 1$) и одна кошечка, которая дошла до холма 2 ($h_1 = 2$) в момент времени 3. Тогда, если кормильщик отправится за кошками от холма 1 в момент времени 2 или 3, то он сможет забрать эту кошку. Но если он отправится от холма 1 в момент времени 1, то он не сможет этого сделать. Если кормильщик отправится за кошкой в момент времени 2, то кошка будет ждать его 0 единиц времени, если же он отправится в момент времени 3, то кошка будет ждать его 1 единицу времени.

Ваша задача — составить расписание отправки от холма 1 для кормильщиков так, чтобы общее время ожидания кошек до того как их заберут было минимальным.

Формат входных данных

В первой строке входных данных содержится три целых числа n, m, p ($2 \leq n \leq 10^5, 1 \leq m \leq 10^5, 1 \leq p \leq 100$).

Во второй строке содержится $n-1$ положительных целых чисел $d_2, d_3, \dots, d_n$ ($1 \leq d_i < 10^4$).

В каждой из следующих m строк содержится по два целых числа h_i и t_i ($1 \leq h_i \leq n, 0 \leq t_i \leq 10^9$).

Формат выходных данных

Выведите целое число, минимальную сумму времен ожидания всех кошек.

Примеры

стандартный ввод	стандартный вывод
4 6 2 1 3 5 1 0 2 1 4 9 1 10 2 10 3 12	3

Задача С. ЗОРП 1

Имя входного файла: `knapsack-1.in`
Имя выходного файла: `knapsack-1.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Перед вами лежат n котиков. Каждый котик характеризуется своим весом w_i и своей мимимишностью c_i . Вы хотите выбрать некоторое число котиков суммарным весом не более чем S так, чтобы их суммарная мимимишность была максимально возможной.

Формат входных данных

В первой строке содержатся два целых числа n и S — число котиков и максимальный допустимый суммарный вес ($1 \leq n \leq 100$, $1 \leq S \leq 10^4$). Следующие n строк содержат по два целых числа w_i и c_i — вес и мимимишность i -го котика ($1 \leq w_i \leq 10^4$, $0 \leq c_i \leq 10^7$).

Формат выходных данных

В первой строке выведите суммарную мимимишность выбранных котиков. Во вторую строку выведите целое число k — количество выбранных котиков. В третьей строке выведите k чисел — номера выбранных котиков. Если оптимальных ответов несколько, то разрешается вывести любой из них.

Примеры

knapsack-1.in	knapsack-1.out
3 10	11
1 2	2
4 3	3 1
8 9	

Задача D. ЗОРП 2

Имя входного файла: `knapsack-2.in`
Имя выходного файла: `knapsack-2.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Перед вами лежат n котиков. Каждый котик характеризуется своим весом w_i и своей мимимишностью c_i . Вы хотите выбрать некоторое число котиков суммарным весом не более чем S так, чтобы их суммарная мимимишность была максимально возможной.

Формат входных данных

В первой строке содержатся два целых числа n и S — число котиков и максимальный допустимый суммарный вес ($1 \leq n \leq 100$, $1 \leq S \leq 10^9$). Следующие n строк содержат по два целых числа w_i и c_i — вес и мимимишность i -го котика ($1 \leq w_i \leq 10^7$, $0 \leq c_i \leq 10^4$). Гарантируется, что сумма всех c_i не превосходит 10^4 .

Формат выходных данных

В первой строке выведите суммарную мимимишность выбранных котиков. Во вторую строку выведите целое число k — количество выбранных котиков. В третьей строке выведите k чисел — номера выбранных котиков. Если оптимальных ответов несколько, то разрешается вывести любой из них.

Примеры

knapsack-2.in	knapsack-2.out
3 10	11
1 2	2
4 3	3 1
8 9	

Задача Е. Распродажа

Имя входного файла: `sale.in`
Имя выходного файла: `sale.out`
Ограничение по времени: 3 секунды
Ограничение по памяти: 256 мегабайт

В супермаркете «На троечку» часто происходят распродажи товаров, срок годности которых подходит к концу. Каждый товар привозят в магазин в определенное время, а через некоторое его вывозят из магазина, в связи с окончанием срока годности. Более формально, каждый товар имеет стоимость c_i , время его завоза в магазин a_i и время его вывоза из магазина b_i .

У Иннокентия есть хитрый план похода в магазин. Даже несколько. Каждый план похода в магазин выглядит так: Иннокентий выбирает какое-то время, когда он появится в магазине m_j , время s_j , которое он проведет в магазине среди огромных стеллажей товаров, и сумму денег k_j , которую он рассчитывает потратить. Для каждого плана он хочет узнать, сможет ли он осуществить его, т. е. верно ли, что он сможет во время своего пребывания в магазине купить несколько товаров суммарной стоимостью **ровно** k_j , при этом все выбранные товары должны быть в магазине на протяжении всего пребывания Иннокентия в магазине.

Помогите Иннокентию определить, какие из его планов можно выполнить.

Формат входных данных

В первой строке входных данных содержится число N — общее количество товаров в магазине ($1 \leq N \leq 800$). Далее содержатся описания товаров, каждый товар описывается тремя целыми числами c_i, a_i, b_i , обозначающими стоимость товара, время его завоза и время его вывоза из магазина ($1 \leq c_i \leq 1\,000, 1 \leq a_i < b_i \leq 10^9$).

Далее содержится число M — количество планов Иннокентия ($1 \leq M \leq 800\,000$). Каждый план описывается тремя целыми числами m_j, k_j, s_j , обозначающими время прихода Иннокентия в магазин, сумму денег, которую он готов потратить в этом плане и длительность его пребывания в магазине ($1 \leq m_j \leq 10^9, 1 \leq k_j \leq 100\,000, 0 \leq s_j \leq 10^9$).

Помните, что это только планы, т. е. ситуация в магазине не меняется вне зависимости от того, может ли Иннокентий осуществить план или нет.

Формат выходных данных

Для каждого плана в отдельной строке выведите «YES», если Иннокентий может его осуществить, и «NO» в противном случае.

Примеры

sale.in	sale.out
5	YES
6 2 7	NO
5 4 9	YES
1 2 4	YES
2 5 8	NO
1 3 9	
5	
2 7 1	
2 7 2	
3 2 0	
5 7 2	
4 1 5	

Задача F. Медианная сумма

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

Вам дан массив из N чисел $A_1, A_2, \dots, A_N$.

Рассмотрим суммы всех непустых подмножеств этого массива. Это $2^N - 1$ сумм, нечетное число.

Пусть $S_1, S_2, \dots, S_{2^N-1}$ это список всех этих сумм в неубывающем порядке. Найдите медиану этого массива, то есть число $S_{2^{N-1}}$.

Формат входных данных

В первой строке входных данных находится число N ($1 \leq N \leq 2000$). В следующей строке находятся элементы массива A_i ($1 \leq A_i \leq 2000$).

Формат выходных данных

Выведите единственное число — медиану отсортированного массива сумм.

Примеры

стандартный ввод	стандартный вывод
3 1 2 1	2
1 58	58

Задача G. Интересные числа

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

Софья считает число интересным, если его цифры идут в неубывающем порядке. Например, числа 123, 1111 или 888999 – интересные.

Софья заинтересовалась, сколько существует интересных положительных чисел, лежащих в диапазоне от L до R включительно. Это число может оказаться довольно большим для больших L и R , поэтому Софья хочет найти остаток от деления этого числа на $10^9 + 7$.

Требуется написать программу, которая по заданным L и R определяет количество интересных чисел, лежащих в диапазоне от L до R включительно, и выводит остаток от деления этого числа на $10^9 + 7$.

Формат входных данных

Входной файл содержит две строки. Первая строка содержит число L , вторая строка содержит число R ($1 \leq L \leq R \leq 10^{100}$).

Формат выходных данных

Выходной файл должен одно целое число — остаток от деления количества интересных чисел, лежащих в диапазоне от L до R включительно, на $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
1 100	54

Задача Н. Петя и прямоугольники

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

Маленький Петя очень любит прямоугольники. Петя дал маме список прямоугольников, которые он хочет получить в подарок на Новый Год. Каждый прямоугольник характеризуется w и высотой h . Мама хочет сделать Пете приятное и купить все прямоугольники из его списка. Мама отправилась в магазин и узнала, что цена одного прямоугольника равна его площади. К ее счастью, в магазине действует предновогодняя акция, позволяющая покупать прямоугольники не по одному, а сразу наборами. Стоимость одного набора равна ширине самого широкого прямоугольника, умноженной на высоту самого высокого прямоугольника из этого набора. Обратите внимание, что поворачивать прямоугольники (тем самым меняя местами ширину и высоту) нельзя. Помогите маме Пети купить все прямоугольники из списка ее сына, потратив на это наименьшее количество денег.

Формат входных данных

В первой строке записано число N ($1 \leq N \leq 200\,000$) — количество прямоугольников в списке Пети. В каждой из следующих N строк записаны по 2 целых положительных числа, не превышающих 10^6 — ширина и высота очередного прямоугольника.

Формат выходных данных

Выведите одно число — наименьшее количество денег, которое может потратить мама чтобы купить Пете все прямоугольники из его списка.

Примеры

стандартный ввод	стандартный вывод
4 100 1 15 15 20 5 1 100	500
5 1 10 2 20 3 30 4 40 10 1	170

Задача I. Рюкзак

Имя входного файла: `knapsack.in`
Имя выходного файла: `knapsack.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 32 мегабайта

Найдите максимальный вес золота, который можно унести в рюкзаке вместительностью S , если есть N золотых слитков с заданными весами.

Формат входных данных

В первой строке входного файла записано одно число — S ($1 \leq S \leq 10^6$).

Далее следует N неотрицательных целых чисел ($1 \leq N \leq 3000$), не превосходящих 10^6 — веса слитков.

При этом N не вводится, вам нужно будет просто считывать все веса, пока не закончатся числа во вводе.

Формат выходных данных

Выведите искомый максимальный вес и на следующей строке веса предметов, с помощью которых можно набрать максимальный вес.

Примеры

<code>knapsack.in</code>	<code>knapsack.out</code>
10 1 4 8	9 8 1
20 5 7 12 18	19 12 7