

Задача А. 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 В. Банкомат

Имя входного файла: `atm.in`
Имя выходного файла: `atm.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 6 мегабайт

В некотором государстве в обращении находятся банкноты определенных номиналов. Национальный банк хочет, чтобы банкомат выдавал любую запрошенную сумму при помощи минимального числа банкнот, считая, что запас банкнот каждого номинала неограничен. Помогите Национальному банку решить эту задачу.

Формат входных данных

Первая строка входных данных содержит натуральное число N не превосходящее 100 — количество номиналов банкнот в обращении. Вторая строка входных данных содержит N различных натуральных чисел x_1, x_2, \dots, x_N , не превосходящих 10^6 — номиналы банкнот. Третья строка содержит натуральное число S , не превосходящее 10^6 — сумму, которую необходимо выдать.

Формат выходных данных

В первую строку выходного файла выведите минимальное число слагаемых (или -1, если такого представления не существует). Во вторую строку выведите это представление в любом порядке.

Примеры

<code>atm.in</code>	<code>atm.out</code>
5 1 3 7 12 32 40	3 1 7 32

Задача С. 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 D. Сокровища

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

Дочь короля Флатландии собирается выйти за прекрасного принца. Принц хочет подарить принцессе сокровища, но он не уверен какие именно бриллианты из своей коллекции выбрать.

В коллекции принца n бриллиантов, каждый характеризуется весом w_i и стоимостью v_i . Принц хочет подарить наиболее дорогие бриллианты, однако король умен и не примет бриллиантов суммарного веса больше R . С другой стороны, принц будет считать себя жадным всю оставшуюся жизнь, если подарит бриллиантов суммарным весом меньше L .

Помогите принцу выбрать набор бриллиантов наибольшей суммарной стоимости, чтобы суммарный вес был в отрезке $[L, R]$.

Формат входных данных

Первая строка содержит число n ($1 \leq n \leq 32$), L и R ($0 \leq L \leq R \leq 10^{18}$). Следующие n строк описывают бриллианты и содержит по два числа — вес и стоимость соответствующего бриллианта ($1 \leq w_i, v_i \leq 10^{15}$).

Формат выходных данных

Первая строка вывода должна содержать k — количество бриллиантов, которые нужно подарить принцессе. Вторая строка должна содержать номера даримых бриллиантов.

Бриллианты нумеруются от 1 до n в порядке появления во входных данных.

Если составить подарок принцессе невозможно, то выведите 0 в первой строке вывода.

Примеры

dowry.in	dowry.out
3 6 8	1
3 10	2
7 3	
8 2	

Задача Е. Погрузка на корабли

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

В порту города S . происходит массовая погрузка контейнеров на корабли. Всего в порту n контейнеров с грузом, i -й из них имеет массу a_i тонн и их все нужно погрузить на корабли. При этом суммарная масса предметов, погруженных на один корабль не должна превышать W тонн.

Корабли по очереди приходят в порт, нагружаются товарами и отплывают. При этом в любой момент времени можно грузить на корабль только самый левый или самый правый из оставшихся контейнеров.

На какое минимальное количество кораблей получится погрузить все контейнеры в порту города S . ?

Формат входных данных

В первой строке вводится два целых положительных числа n ($1 \leq n \leq 4000$) и W ($1 \leq W \leq 10^{18}$) — количество контейнеров и грузоподъемность кораблей.

Во второй строке содержатся n целых положительных чисел a_i ($1 \leq a_i \leq \min(10^9, W)$) — массы контейнеров.

Формат выходных данных

Выведите одно целое число k — минимальное количество кораблей, на которое можно погрузить все контейнеры.

Примеры

стандартный ввод	стандартный вывод
7 10 4 5 1 1 3 2 4	2
7 4 3 1 2 1 1 3 1	3

Задача F. 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 G. Математическая собака

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Сеня готовится к математической абаке, которая пройдёт завтра утром. Это интеллектуальная игра, где надо решать задачи и за правильные решения получать очки. Выигрывает команда, которая наберёт наибольшее число очков.

Сеня решил участвовать в одиночку, поэтому, чтобы победить, ему придётся хорошо планировать свои действия. Для начала он узнал точные правила игры. В этой версии математической абаки все задачи расставлены в таблицу $m \times n$ (состоящую из m строк и n столбцов). Каждый столбец соответствует какой-то математической теме (например, какой-то столбец может состоять из m геометрических задач, другой — из m задач по теории чисел, и так далее). Внутри столбца задачи отсортированы сверху вниз по сложности. Таким образом, каждая задача имеет сложность, задающуюся номером строки от 1 до m , и тему, задающуюся номером столбца от 1 до n . За правильное решение задачи сложности номер i из темы номер j команда получает $10 \cdot i$ очков. Кроме того, если правильно решить все задачи из одной темы (то есть правильно решить все задачи из какого-то столбца), то за каждую такую тему дополнительно начисляется по P очков. (В некоторых версиях абаки бонусы начисляются также за полностью решённые строчки, но в той абаке, в которой завтра поучаствует Сеня, такой бонус не предусмотрен.)

Сеня — очень опытный игрок, поэтому он заранее знает, сколько времени ему потребуется на каждую клетку таблицы: он может найти и сдать правильный ответ к задаче i -й сложности на j -ю тему ровно за $t_{i,j}$ секунд. При этом для вашего удобства Сеня смог переупорядочить темы так, что чем больше номер темы, тем сложнее эта тема даётся Сене. Формально это означает, что в рамках одной и той же сложности время, затрачиваемое на решение задачи этой сложности, не убывает с увеличением номера темы:

$$j < k \text{ влечёт, что } t_{i,j} \leq t_{i,k}.$$

При этом не гарантируется, что время на решение задачи возрастает с номером строки: возможно, задачу, которую жюри абаки считает сложной, Сеня решит в два счёта, а та, которую жюри считает простенькой, займёт у Сени час.

Сеня заранее выберет клетки, задачи из которых он решит на абаке. При этом суммарное время, требуемое на эти задачи, не должно превосходить общую продолжительность игры, равную T секундам. Вычислите, какое наибольшее число очков сможет набрать Сеня на игре.

Формат входных данных

В первой строке находится четыре целых числа m, n, T, P разделённых пробелами — количество сложностей, количество тем задач, длина математической абаки в секундах и бонус за решение всех задач из одной темы ($1 \leq m, n \leq 100, 0 \leq T, P \leq 10\,000$).

В следующих m строках задано время, затрачиваемое на решение задач абаки: в i -й из них находятся целые числа $t_{i,1}, t_{i,2}, \dots, t_{i,n}$ — количество секунд, требуемое на решение задачи i -й сложности из первой, второй, \dots , n -й темы. Гарантируется, что каждая из этих m последовательностей неубывающая ($0 \leq t_{i,1} \leq t_{i,2} \leq \dots \leq t_{i,n} \leq 10\,000$).

Формат выходных данных

Выведите одно целое неотрицательное число — максимальное число очков, которое сможет набрать Сеня на абаке (с учётом бонусов за решение всех задач из столбца).

Примеры

стандартный ввод	стандартный вывод
2 2 7 932 9 10 7 11	20
4 4 30 10000 1 1 1 1 1 1 1 1 1 1 1 1 27 31 10000 10000	10100
4 4 30 10000 1 1 1 1 1 1 1 1 1 1 1 1 28 31 10000 10000	240
4 4 30 100 1 1 1 1 1 1 1 1 1 1 1 1 27 31 10000 10000	240
1 8 15 90 1 3 5 7 9 11 13 15	300
1 8 16 90 1 3 5 7 9 11 13 15	400