

Сортировка-2. Применение.

А. Дан массив слов, содержащих строчные латинские буквы. Упорядочить слова по возрастанию с учётом обратного чтения, т.е. не по начальным, а по конечным буквам (упорядоченный таким образом словарь называется *обратным* и удобен при изучении особенностей строения конца слов и вообще словообразования, подборе рифм).

В единственной строке вводится N ($1 \leq N \leq 10^3$) слов из строчных букв длиной не более 50 символов, разделённых пробелами.

Input	Output
window table ten screen	table screen ten window

Определение. Сортировка называется *стабильной*, если она не меняет местами равные элементы исходного массива.

При первом прочтении определения может возникнуть впечатление, что оно имеет мало смысла. Действительно, ведь равные элементы неразличимы и нет никакой разницы, в каком порядке они идут. Это совсем не так. Достаточно хорошей иллюстрацией могут служить электронные таблицы (Excel, GoogleSheets, и проч.). Представьте таблицу с двумя столбцами, в первом записаны фамилии учеников, во втором класс, в котором учится каждый из учеников. Мы хотим расположить строки этой таблицы так, чтобы сначала были все ученики первого класса в алфавитном порядке, затем второго, также в алфавитном порядке, и так далее. Для этого нам достаточно сначала отсортировать таблицу по первому столбцу, а затем по второму. Благодаря тому, что в таблицах используется стабильная сортировка, после того, как мы отсортировали учеников по классам, среди равных элементов (учеников из одного класса) исходный порядок (алфавитный) не нарушится.

В. *Благосостояние и стабильность.*

В некоей стране есть три разных валюты — динары, лиры и тугрики. Известно, что по текущему курсу валют, одна лира равна пяти динарам, а один тугрик равен семи динарам. Журнал «Благосостояние и стабильность» каждый год публикует список самых состоятельных граждан (не более 1000). Главный казначей страны выдает редактору журнала список главных богачей в следующем формате:

<фамилия> <количество динаров> <количество лир> <количество тугриков>.

Список составлен в порядке влиятельности богачей. Ваша задача отсортировать список по общему благосостоянию, причем если двое граждан имеют одинаковое количество денег, в вашем списке они должны быть расположены в порядке влиятельности (сортировка должна быть *стабильной*, стандартная сортировка в питоне таковой является).

В первой строке задается число N — количество людей в списке казначея ($N \leq 1000$). В последующих N строках задается список казначея в описанном формате.

Выведите список фамилий самых богатых людей. Каждую фамилию нужно выводить на отдельной строке.

Input	Output
4	Rosenblum
Ivanov 34 0 0	Petrov
Petrov 0 7 0	Sidorov
Sidorov 0 0 5	Ivanov
Rosenblum 1000 1000 100500	

С. Результаты олимпиады

Во время проведения олимпиады каждый из участников получил свой идентификационный номер — целое неотрицательное число. Необходимо отсортировать список участников олимпиады по количеству набранных ими баллов.

На первой строке дано число N ($1 \leq N \leq 1000$) — количество участников. В следующих N строках даны идентификационный номер и набранное число баллов соответствующего участника. Во входных данных могут встретиться строки с совпадающими идентификационными номерами. Все числа во входном файле не превышают 10^5 .

В выходной файл выведите исходный список в порядке убывания баллов. Если у некоторых участников одинаковые баллы, то их между собой нужно упорядочить в порядке возрастания идентификационного номера.

Input	Output
3	305 90
101 80	101 80
305 90	200 14
200 14	
3	25 90
20 80	30 90
30 90	20 80
25 90	

Д. Иннокентий решает задачи

Мальчик Иннокентий решает вступительную работу в летний математический лагерь. В ней N заданий, которые можно выполнять в произвольном порядке. Разные задачи требуют разного времени для решения. При этом известно, что если задание с номером i выполнять j -м по счету, Иннокентию потребуется $T_i \times j$ времени: чем больше думаешь, тем больше устаешь. Например, если начать с первой задачи, а затем выполнить вторую, то потребуется $T_1 \times 1 + T_2 \times 2$ времени, а если выполнить сначала вторую задачу, а затем первую — то $T_2 \times 1 + T_1 \times 2$.

Подскажите Иннокентию, в каком порядке нужно решать задачи, чтобы на выполнение всей работы ушло как можно меньше времени.

В первой строке вводится число N , во второй строке — N чисел через пробел T_1, T_2, \dots, T_N , разделённые пробелами. Все числа целые и удовлетворяют следующим ограничениям: $0 < N \leq 10, 0 < T_i \leq 100$.

Требуется вывести сначала минимальное время, за которое можно решить все задачи, а затем — номера задач в том порядке, в котором их нужно решать, чтобы уложиться в это время. Все числа разделяются пробелами. Если решений несколько, нужно выдать любое из них.

Input	Output
2	7
2 3	2 1

Е. Такси

После затянувшегося совещания директор фирмы решил заказать такси, чтобы развезти сотрудников по домам. Он заказал N машин — ровно столько, сколько у него сотрудников. Однако когда они подъехали, оказалось, что у каждого водителя такси свой тариф за 1 километр.

Директор знает, какому сотруднику сколько километров от работы до дома (к сожалению, все сотрудники живут в разных направлениях, поэтому нельзя отправить двух сотрудников на одной машине). Теперь директор хочет определить, какой из сотрудников на каком такси должен поехать домой, чтобы суммарные затраты на такси (а их несет фирма) были минимальны.

Первая строка входных данных содержит натуральное число N ($1 \leq N \leq 1000$) — количество сотрудников компании (совпадающее с количеством вызванных машин такси). Далее записано N чисел, задающих расстояния в километрах от работы до домов сотрудников компании (первое число — для первого сотрудника, второе — для второго и т.д.). Все расстояния — положительные целые числа, не превышающие 1000. Далее записано еще N чисел — тарифы за проезд одного километра в такси (первое число — в первой машине такси, второе — во второй и т.д.). Тарифы выражаются положительными целыми числами, не превышающими 10000.

В первой строке программа должна вывести минимальную стоимость доставки всех сотрудников. Во второй строке программа должна вывести N чисел. Первое число — номер такси, в которое должен сесть первый сотрудник, второе число — номер такси, в которое должен сесть второй и т.д., чтобы суммарные затраты на такси были минимальны. Если вариантов раскладки сотрудников, при которых затраты минимальны, несколько, выведите любой из них.

Input	Output
3 10 20 30 50 20 30	1700 1 3 2
5 10 20 1 30 30 3 3 3 2 3	243 5 1 3 2 4

Ф. *Решение задач*

Вася готовится к олимпиаде. Учитель дал ему N ($1 \leq N \leq 100000$) задач для тренировки. Для каждой из этих задач известно, каким умением a_i нужно обладать для её решения. Это означает, что если текущее умение Васи больше либо равно заданного умения для задачи, то он может ее решить.

Кроме того, после решения i -й задачи Васино умение увеличивается на число b_i .

Исходное умение Васи равно A . Решать данные учителем задачи он может в произвольном порядке. Какое максимальное количество задач он сможет решить, если выберет самый лучший порядок их решения?

Сначала вводятся два целых числа N, A ($1 \leq N \leq 100000, 0 \leq A \leq 10^9$) — количество задач и исходное умение. Далее идут N пар целых чисел a_i, b_i ($1 \leq a_i \leq 10^9, 1 \leq b_i \leq 10^9$) — соответственно сколько умения нужно для решения i -й задачи и сколько умения прибавится после её решения. Выведите одно число — максимальное количество задач, которое Вася сможет решить.

Input	Output
3 2 3 1 2 1 1 1	3

Г. *Закраска прямой*

На числовой прямой окрасили N отрезков. Известны координаты левого и правого концов каждого отрезка (L_i и R_i). Найти длину окрашенной части числовой прямой.

В первой строке находится число N , в следующих N строках — пары L_i и R_i . L_i и R_i — целые, $-10^9 \leq L_i \leq R_i \leq 10^9, 1 \leq N \leq 15000$.

Вывести одно число — длину окрашенной части прямой.

Input	Output
1 10 10	0
2 10 20 20 40	30

Н. *Белоснежка и N гномов*

У Белоснежки N гномов, и все они очень разные. Она знает, что для того, чтобы уложить спать i -го гнома нужно a_i минут, и после этого он будет спать ровно b_i минут. Помогите Белоснежке узнать, может ли она получить хотя бы минутку отдыха, когда все гномы будут спать, и если да, то в каком порядке для этого нужно укладывать гномов спать.

Например, пусть есть всего два гнома, $a_1 = 1, b_1 = 10, a_2 = 10, b_2 = 20$. Если Белоснежка сначала начнет укладывать первого гнома, то потом ей потребуется целых 10 минут, чтобы уложить второго, а за это время проснется первый. Если же она начнет со второго гнома, то затем она успеет уложить первого и получит целых 10 минут отдыха.

Первая строка входного файла содержит число N ($1 \leq N \leq 10^5$), вторая строка содержит числа a_1, a_2, \dots, a_n , третья — числа b_1, b_2, \dots, b_n ($1 \leq a_i, b_i \leq 10^9$).

Выведите в выходной файл N чисел — порядок, в котором нужно укладывать гномов спать. Если Белоснежке отдохнуть не удастся, выведите число -1 .

Input	Output
2 1 10 10 20	2 1
2 10 10 10 10	-1