

Задача А. Заезд в ЛКШ

Имя входного файла: `arrival.in`
Имя выходного файла: `arrival.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Знаете ли вы, как непросто организовать заезд в ЛКШ? Например, в 2012 году нужно было заказать автобусы для целых n ЛКШат, мечтающих добраться в «Берендеевы поляны» из Москвы. Один из директоров ЛКШ сообщил другому директору, что можно заказать некоторые из m автобусов. Он узнал вместимость каждого автобуса и сразу понял, какое минимальное количество автобусов ему нужно заказать, чтобы привезти в лагерь всех ЛКШат. А сможете ли вы так же быстро решить эту задачу?

Формат входных данных

В первой строке через пробел записаны целые числа n и m ($1 \leq n \leq 10^6$; $1 \leq m \leq 1000$). В следующей строке через пробел записаны m целых чисел в пределах от 1 до 1000 — вместимости автобусов.

Формат выходных данных

В первой строке выведите число k — минимальное количество автобусов, которое придётся заказать директору. В следующей строке выведите через пробел k целых чисел — номера автобусов, которые нужно заказать. Автобусы пронумерованы от 1 до m в том порядке, в которых они перечислены во входных данных. Если возможных решений несколько, выведите любое. Если решения нет, в единственной строке выведите “-1”.

Примеры

<code>arrival.in</code>	<code>arrival.out</code>
345 5 100 130 190 140 150	3 1 3 4
345 3 100 100 100	-1

Задача В. Тестирующая система

Имя входного файла: `ejudge.in`
Имя выходного файла: `ejudge.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Юный программист Саша написал свою первую тестирующую систему. Он так обрадовался тому, что она скомпилировалась, что решил пригласить школьных друзей на свой собственный констест.

Но в конце тура выяснилось, что система не умеет сортировать команды в таблице результатов. Помогите Саше реализовать эту сортировку.

Команды упорядочиваются по правилам ACM:

- по количеству решённых задач в порядке убывания;
- при равенстве количества решённых задач — по штрафному времени в порядке возрастания;
- при прочих равных — по номеру команды в порядке возрастания.

Формат входных данных

Первая строка содержит натуральное число n ($1 \leq n \leq 100\,000$) — количество команд, участвующих в констесте. В i -й из следующих n строк записано количество решённых задач S ($0 \leq S \leq 100$) и штрафное время T ($0 \leq T \leq 100\,000$) команды с номером i .

Формат выходных данных

В выходной файл выведите n чисел — номера команд в отсортированном порядке.

Примеры

<code>ejudge.in</code>	<code>ejudge.out</code>
5	5 2 1 3 4
3 50	
5 720	
1 7	
0 0	
8 500	

Задача С. Дареный конь

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

Никита обожает стабильность. Вчера ночью Гриша подарил ему массив пар целых чисел.

Гриша крайне щедрый молодой человек, поэтому массив оказался несказанно огромным. Никита сделал вид, что доволен подарком, однако, как можно быть довольным неотсортированным массивом?!

Никита хочет отсортировать массив. А вы же помните, что Никита обожает стабильность? Поэтому он хочет, чтобы его массив был стабильно отсортирован по первому числу из пары (Пара (a_i, b_i) считается меньше пары (a_j, b_j) , если $a_i < a_j$ или $a_i = a_j$ и при этом $i < j$). Но еще больше Никита любит полдник. А еще когда ему делают приятно. Пока он сходит до столовой и обратно, сделайте ему приятно, стабильно отсортируйте его массив.

Формат входных данных

В первой строке находится одно натуральное число n — размер подаренного массива ($1 \leq n \leq 50\,000$).

В следующих n строках заданы по два натуральных числа a_i, b_i — пары чисел в массиве ($1 \leq a_i \leq 10, 1 \leq b_i \leq 10\,000$).

Формат выходных данных

Выведите в n строках стабильно отсортированный массив.

Примеры

<code>count.in</code>	<code>count.out</code>
6	1 6
3 1	2 3
2 3	2 4
1 6	2 2
2 4	3 1
3 3	3 3
2 2	

Задача D. Число

Имя входного файла: `number.in`
Имя выходного файла: `number.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Вася написал на длинной полоске бумаги большое число и решил похвастаться своему старшему брату Пете этим достижением. Но только он вышел из комнаты, чтобы позвать брата, как его сестра Катя вбежала в комнату и разрежала полоску бумаги на несколько частей. В результате на каждой части оказалось одна или несколько идущих подряд цифр.

Теперь Вася не может вспомнить, какое именно число он написал. Только помнит, что оно было очень большое. Чтобы утешить младшего брата, Петя решил выяснить, какое максимальное число могло быть написано на полоске бумаги перед разрезанием. Помогите ему!

Формат входных данных

Входной файл содержит одну или более строк, каждая из которых содержит последовательность цифр. Количество строк во входном файле не превышает 100, каждая строка содержит от 1 до 100 цифр. Гарантируется, что хотя бы в одной строке первая цифра отлична от нуля.

Формат выходных данных

Выведите в выходной файл одну строку — максимальное число, которое могло быть написано на полоске перед разрезанием.

Примеры

<code>number.in</code>	<code>number.out</code>
2 20 004 66	66220004
3	3

Задача Е. K -ая порядковая статистика

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

На уроке физкультуры преподаватель заинтересовался, кто из школьников — k -й по росту. Помогите ему определить рост этого школьника.

Формат входных данных

Во входном файле две строки. В первой строке содержится число k . Во второй строке через пробел записаны целые числа a_i , соответствующие росту школьников. Гарантируется, что школьников не более 10^5 , а рост каждого школьника положителен и не превышает 10^{30} .

Формат выходных данных

Выведите одно число — ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
4 1 2 3 4 5 6 7 8 9 10	7

Задача F. Количество инверсий

Имя входного файла: `inverse.in`
Имя выходного файла: `inverse.out`
Ограничение по времени: 2 секунды
Ограничение по памяти: 64 мегабайта

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 $a_i > a_j$.

Формат входных данных

Первая строка входного файла содержит натуральное число n ($1 \leq n \leq 80\,000$) — количество элементов массива. Вторая строка содержит n попарно различных элементов массива A — целых неотрицательных чисел, не превосходящих 10^6 .

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
8 999994 999989 999982 999972 999969 999961 999954 999950	28