

Содержание

02.Base [3/3]	3
Задача 02A. Заезд в ЛКШ [0.5 sec, 256 mb]	3
Задача 02B. Тестирующая система [0.5 sec, 256 mb]	4
Задача 02C. Скалярное произведение [0.5 sec, 256 mb]	5
02.Advanced [4/6]	6
Задача 02D. Число [0.5 sec, 256 mb]	6
Задача 02E. Белоснежка и n гномов [0.5 sec, 256 mb]	7
Задача 02F. Точки и отрезки [0.5 sec, 256 mb]	8
Задача 02G. Отрезки [0.5 sec, 256 mb]	9
Задача 02H. Том Сойер и его друзья [0.5 sec, 256 mb]	10
Задача 02I. $F^2 \parallel C_{max}$ [0.3 sec, 256 mb]	11
02.Hard [0/2]	12
Задача 02K. Коробки [0.5 sec, 256 mb]	12
Задача 02L. Эльфы и олени [0.5 sec, 256 mb]	13

Общая информация:

Вход в констест: <http://contest.yandex.ru/contest/2793/>

Дедлайн на задачи: 9 дней, до 2016-09-24 23:59.

К каждой главе есть более простые задачи (base), посложнее (advanced), и сложные (hard).

В скобках к каждой главе написано сколько любых задач из этой главы нужно сдать.

Сайт курса: <https://compscicenter.ru/courses/algorithms-1/2016-autumn/>

Семинары ведут Сергей Копелиович (burunduk30@gmail.com, vk.com/burunduk1) и Алексей Кладов (aleksey.kladov@gmail.com).

В каждом условии указан таймлимит для C/C++.

Таймлимит для Java примерно в 2-3 раза больше.

Таймлимит для Python примерно в 6 раз больше.

C++:

Быстрый ввод-вывод.

http://acm.math.spbu.ru/~sk1/algo/input-output/fread_write_export.cpp.html Более подробно про ввод-вывод.

http://acm.math.spbu.ru/~sk1/algo/input-output/cpp_common.html

В некоторых задачах нужен STL, который активно использует динамическую память (set-ы, map-ы) переопределение стандартного аллокатора ускорит вашу программу:

<http://acm.math.spbu.ru/~sk1/algo/memory.cpp.html>

Java:

Быстрый ввод-вывод.

http://acm.math.spbu.ru/~sk1/algo/input-output/java/java_common.html

02.Base [3/3]

Задача 02А. Заезд в ЛКШ [0.5 sec, 256 mb]

Знаете ли вы, как непросто организовать заезд в ЛКШ? Например, в 2010 году нужно было заказать автобусы для целых n ЛКШат, мечтающих добраться в “Орлёнок” из Саратова. Директору смены сообщили, 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

arrival.in	arrival.out
345 5 100 130 190 140 150	3 1 3 4
345 3 100 100 100	-1

Задача 02В. Тестирующая система [0.5 sec, 256 mb]

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

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

Команды упорядочиваются по правилам АСМ:

- 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 контексте. 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.

Пример

ejudge.in	ejudge.out
5	5 2 1 3 4
3 50	
5 720	
1 7	
0 0	
8 500	

Задача 02С. Скалярное произведение [0.5 sec, 256 mb]

Даны два вектора: $v_1 = (x_1, x_2, \dots, x_n)$ и $v_2 = (y_1, y_2, \dots, y_n)$. Скалярным произведением этих векторов называется значение, вычисляемое по формуле: $x_1y_1 + x_2y_2 + \dots + x_ny_n$.

Разрешено переставлять координаты каждого из векторов любым образом. Выберите такие их перестановки, чтобы скалярное произведение двух полученных векторов было минимальным и выведите его значение.

$1 \leq n \leq 800$. $-100\,000 \leq x_i, y_i \leq 100\,000$.

Формат входных данных

Первая строка входного файла содержит единственное целое число t — количество наборов тестовых данных. Далее следуют сами наборы, по три строки в каждом. Первая строка тестового набора содержит единственное целое число n . Две следующие строки содержат по n целых чисел, задающих координаты соответствующего вектора, каждая.

Формат выходных данных

Для каждого набора выведите строку с номером этого набора и ответом на задачу — значением минимального скалярного произведения. Следуйте формату, указанному в примере.

Пример

product.in	product.out
2	Case #1: -25
3	Case #2: 6
1 3 -5	
-2 4 1	
5	
1 2 3 4 5	
1 0 1 0 1	

02.Advanced [4/6]

Задача 02D. Число [0.5 sec, 256 mb]

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

number.in	number.out
2 20 004 66	66220004
3	3

Задача 02Е. Белоснежка и n гномов [0.5 sec, 256 mb]

«Ну не гномы, а наказание какое-то!», — подумала Белоснежка, в очередной раз пытаясь уложить гномов спать. Одного уложишь — другой уже проснулся! И так всю ночь.

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 9 минут отдыха.

Формат входных данных

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 ($1 \leq a_i, b_i \leq 10^9$).

Формат выходных данных

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

Пример

dwarfs.in	dwarfs.out
2 1 10 10 20	2 1
2 10 10 10 10	-1
3 1 4 1 5 3 4	2 1 3

Задача 02F. Точки и отрезки [0.5 sec, 256 mb]

Дано n отрезков на числовой прямой и m точек на этой же прямой. Для каждой из данных точек определите, скольким отрезкам она принадлежит. Точка x считается принадлежащей отрезку с концами a и b , если выполняется двойное неравенство $\min(a, b) \leq x \leq \max(a, b)$.

Формат входных данных

Первая строка содержит два целых числа n ($1 \leq n \leq 10^5$) — число отрезков и m ($1 \leq m \leq 10^5$) — число точек. В следующих n строках записаны по два целых числа a_i и b_i — координаты концов соответствующего отрезка. В последней строке записаны m целых чисел — координаты точек. Все числа во входном файле не превосходят по модулю 10^9 .

Формат выходных данных

В выходной файл выведите m чисел — для каждой точки выведите количество отрезков, в которых она содержится.

Примеры

segments.in	segments.out
2 2 0 5 7 10 1 6	1 0
1 3 -10 10 -100 100 0	0 0 1

Задача 02G. Отрезки [0.5 sec, 256 mb]

Даны отрезки на прямой. Какое максимальное количество отрезков можно выбрать так, чтобы никакие два из них не пересекались? Отрезки считаются открытыми.

Формат входных данных

В первой строке входного файла задано целое число n ($1 \leq n \leq 100\,000$). В следующих n строках описаны отрезки; i -я из этих строк содержит два целых числа l_i и r_i через пробел — координаты начала и конца отрезка ($1 \leq l_i < r_i \leq 10^9$).

Формат выходных данных

В первой строке выходного файла выведите одно число — максимальное количество выбранных отрезков.

Примеры

segments.in	segments.out
2 1 2 3 5	2
2 1 3 2 6	1
3 2 4 3 5 1 3	2

Задача 02Н. Том Сойер и его друзья [0.5 sec, 256 mb]

Друзья Тома Сойера по очереди красят забор разными красками. Каждый из них красит несколько идущих подряд секций забора в определенный цвет, при этом используемые цвета могут повторяться. Новая краска ложится поверх старой. Для каждой краски вычислите количество секций, которые будут покрашены этой краской после того, как все друзья закончат работу.

Формат входных данных

В первой строке входного файла содержатся два целых числа: N ($1 \leq N \leq 10^9$) и K ($1 \leq K \leq 50\,000$) — количество секций в заборе и количество различных красок соответственно.

Во второй строке содержится единственное число M ($0 \leq M \leq 50\,000$) — количество друзей Тома Сойера.

Далее следуют M строк: в i -ой строке содержится информация о работе друга, который красил забор i -ым по счету, а именно 3 целых числа c_i, l_i, r_i ($1 \leq c_i \leq K, 1 \leq l_i \leq r_i \leq N$) — номер краски, которую использовал i -й друг, номер первой и номер последней покрашенной секции соответственно.

Формат выходных данных

Выведите в единственную строку выходного файла K целых чисел: i -ое число должно быть равно количеству секций, покрашенных i -й краской.

Пример

paint.in	paint.out
5 3 4 1 3 4 2 4 5 3 2 3 1 5 5	1 1 2
5 3 3 1 1 5 2 2 4 1 3 3	3 2 0

Задача 021. F2 || C_{max} [0.3 sec, 256 mb]

Имеется множество из n работ и два станка. Время выполнения i -й работы на первом станке равно a_i , время выполнения i -й работы на втором станке равно b_i . Каждую работу надо выполнить сначала на первом станке, потом на втором. И на первом, и на втором станке работы можно выполнять в произвольном порядке. Каждый станок в каждый момент времени может выполнять только одну работу.

Минимизируйте C_{max} — время выполнения последней работы на втором станке.

Формат входных данных

В первой строке дано одно целое число n ($1 \leq n \leq 100\,000$) — количество работ. В следующей строке n целых чисел от 0 до 10^6 — время выполнения работ на первом станке. В следующей строке n целых чисел от 0 до 10^6 — время выполнения работ на втором станке.

Формат выходных данных

В первой строке выведите единственное число — C_{max} . Во второй строке выведите перестановку — порядок выполнения работ на первом станке. В третьей строке выведите перестановку — порядок выполнения работ на втором станке.

Примеры

f2cmax.in	f2cmax.out
3	16
1 2 3	1 3 2
5 5 5	1 2 3
2	6
3 2	2 1
1 3	2 1

02.Hard [0/2]

Задача 02К. Коробки [0.5 sec, 256 mb]

У Васи в комнате очень много коробок, которые валяются в разных местах. Васина мама хочет, чтобы он прибрался. Свободного места в комнате мало и поэтому Вася решил собрать все коробки и поставить их одну на другую.

К сожалению, это может быть невозможно. Например, если на картонную коробку с елочными украшениями положить что-то железное и тяжелое, то вероятно следующий Новый год придется встречать с новыми игрушками.

Вася взвесил каждую коробку и оценил максимальный вес который она может выдержать. Помогите ему определить какое наибольшее количество коробок m он сможет поставить одну на другую так, чтобы для каждой коробки было верно, что суммарный вес коробок сверху не превышает максимальный вес, который она может выдержать.

Формат входных данных

Первая строка входного файла содержит целое число n ($1 \leq n$) — количество коробок в комнате. Каждая следующая из n строк содержит два целых числа w_i и c_i ($1 \leq w_i \leq 10^5, 1 \leq c_i \leq 10^9$), где w_i — это вес коробки с номером i , а c_i — это вес который она может выдержать.

Формат выходных данных

В выходной файл выведите одно число — ответ на задачу.

Подзадача 1 (50 баллов) $n \leq 1250$

Подзадача 2 (50 баллов) $n \leq 100\,000$

Пример

boxes.in	boxes.out
3 10 11 20 100 30 10	3
3 11 11 20 100 30 10	2

Задача 02L. Эльфы и олени [0.5 sec, 256 mb]

Скоро новый год и Санта-Клаус уже начал готовить свою волшебную оленью упряжку, на которой он развозит подарки детям. Известно, что упряжку везут несколько волшебных оленей, на каждом из которых едут два эльфа.

Но волшебные олени — строптивые животные, поэтому не любые два эльфа могут ехать на любом олене. А именно, каждый олень характеризуется некоторой строптивостью a_i , а каждый эльф — темпераментом b_i . Два эльфа j и k могут ехать на i -м олене в том и только в том случае, если либо $b_j < a_i < b_k$, либо $b_k < a_i < b_j$.

Чтобы его появление было максимально зрелищным, Санта-Клаус хочет, чтобы в его упряжке было как можно больше оленей. Про каждого оленя Санта знает его строптивость, а про каждого эльфа — его темперамент.

Помогите Санте выяснить, какое максимальное количество оленей он сможет включить в упряжку, каких оленей ему следует выбрать, и какие эльфы должны на них ехать.

Формат входных данных

Первая строка входного файла содержит два целых числа m и n — количество оленей и эльфов, соответственно ($1 \leq m, n \leq 100\,000$).

Вторая строка содержит m целых чисел a_i — строптивость оленей ($0 \leq a_i \leq 10^9$). Третья строка содержит n целых чисел b_i — темперамент эльфов ($0 \leq b_i \leq 10^9$).

Формат выходных данных

На первой строке выходного файла выведите одно число k — максимальное количество оленей, которое Санта-Клаус может включить в свою упряжку. На следующих k строках выведите по три целых числа: $d_i, e_{i,1}, e_{i,2}$ — для каждого оленя в упряжке выведите его номер и номера эльфов, которые на нем поедут. Если решений несколько, выведите любое. Выводить эльфов нужно в таком порядке, что $b_{e_{i,1}} < a_{d_i} < b_{e_{i,2}}$.

И эльфы, и олени пронумерованы, начиная с единицы, в том порядке, в котором они заданы во входном файле.

Пример

elves.in	elves.out
4 6	2
2 3 4 5	1 1 2
1 3 2 2 5 2	3 4 5