

Задача А. А+В

Имя входного файла: `aplusb.in`
Имя выходного файла: `aplusb.out`
Ограничение по времени: 1 second
Ограничение по памяти: 64 megabytes

Для начала найдите сумму двух целых чисел.

Формат входных данных

`aplusb.in` или стандартный ввод

Формат выходных данных

`aplusb.out` или стандартный вывод

Примеры

<code>aplusb.in</code>	<code>aplusb.out</code>
7 4	11

Задача В. Мутанты

Имя входного файла: `mutants.in`
Имя выходного файла: `mutants.out`
Ограничение по времени: 4 секунды
Ограничение по памяти: 128 мегабайта

Уже долгое время в Институте Искусств, Мутантов и Информационных Технологий разводят милых разноцветных зверюшек. Для удобства каждый цвет обозначен своим номером, всего цветов не более 10^9 . В один из прекрасных дней в питомнике случилось чудо: все зверюшки выстроились в ряд в порядке возрастания цветов. Пользуясь случаем, лаборанты решили посчитать, сколько зверюшек разных цветов живет в питомнике, и, по закону жанра, попросили вас написать программу, которая поможет им в решении этой нелегкой задачи.

Формат входных данных

В первой строке входного файла содержится единственное число N ($0 \leq N \leq 10^5$) — количество зверюшек в Институте. В следующей строке находятся N упорядоченных по неубыванию неотрицательных целых чисел, не превосходящих 10^9 и разделенных пробелами — их цвета. В третьей строке файла записано число M ($1 \leq M \leq 100\,000$) — количество запросов вашей программе, в следующей строке через пробел записаны M целых неотрицательных чисел (не превышающих $10^9 + 1$).

Формат выходных данных

Выходной файл должен содержать M строчек. Для каждого запроса выведите число зверюшек заданного цвета в питомнике.

Примеры

<code>mutants.in</code>	<code>mutants.out</code>
10	1
1 1 3 3 5 7 9 18 18 57	2
5	1
57 3 9 1 179	2
	0

Задача С. 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 ($1 \leq n \leq 50\,000$) — количество элементов массива. Вторая строка содержит n попарно различных элементов массива A .

Формат выходных данных

В выходной файл выведите одно число — ответ на задачу.

Примеры

<code>inverse.in</code>	<code>inverse.out</code>
4 1 2 4 5	0
4 5 4 2 1	6

Задача D. K -й минимум — 2

Имя входного файла: `kth2.in`
Имя выходного файла: `kth2.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 64 мегабайта

Напишите программу, которая находит k -е в возрастающем порядке число в массиве $A = \langle a_1, a_2, \dots, a_n \rangle$.

Массив A задается с помощью полинома $P(x) = 131x^3 + 77x^2 + 1345x + 1577$: $a_i = P(i) \bmod 2^{32}$.

Формат входных данных

Входной файл содержит два натуральных числа n и k ($1 \leq k \leq n \leq 10^7$).

Формат выходных данных

В выходной файл выведите одно число — ответ на задачу.

Примеры

<code>kth2.in</code>	<code>kth2.out</code>
1 1	3130
10 1	3130

Замечание

Для вычислений по модулю 2^{32} используйте тип `unsigned int` в C/C++ или `LongWord` в Delphi, игнорируя переполнения.

Задача E. +- машина

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

Ваша задача — промоделировать работу +- машины. У машины есть один вход, на который каждую секунду подается целое число A . Если A положительно, то оно добавляется в список чисел, хранящихся в памяти машины. Если же A отрицательно, то машина находит $-A$ -е по возрастанию число из хранящихся в ее памяти и выводит в выходной поток. Выведенное число из памяти удаляется. Числа в памяти хранятся с учетом повторений, например, если в памяти хранились числа 2, 2 и 3, то 3-м по возрастанию числом будет число 3.

Кроме того, если на вход машине подается число 0, то машина выводит все содержимое своей памяти в возрастающем порядке, и память машины очищается.

Формат входных данных

в первой строке содержится количество операций, которые необходимо промоделировать ($1 \leq n \leq 500\,000$). В последующих n строках содержатся сами операции — числа, не превосходящие 10^6 по модулю. Гарантируется, что все запросы корректны.

Изначально память машины пуста.

Формат выходных данных

Выведите в выходной файл числа в том порядке, в котором их выведет +- машина. Разделяйте числа пробелами или переводами строки.

Примеры

стандартный ввод	стандартный вывод
2 1 -1	1
5 1 2 3 -2 -2	2 3

Задача F. Замкнутая ломаная

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

У Пети на листке бумаги нарисованы n точек. Ему необходимо во что бы то ни стало соединить их замкнутой ломаной без самопересечений и самокасаний. Возможно ли это сделать, и если да, то как их нужно соединять?

Формат входных данных

В первой строке входного файла содержится количество точек n ($3 \leq n \leq 131072$). В n последующих строках содержатся координаты точек $x_i y_i$. Все координаты — целые неотрицательные числа, меньшие 10 000. Гарантируется, что среди точек нет двух совпадающих

Формат выходных данных

Если требуемое соединение произвести невозможно, то выведите в выходной файл единственное число -1 . Иначе выведите n строчек, каждая из которых содержит координаты вершин искомой ломаной в порядке обхода. Ломаная не должна иметь самопересечений или самокасаний, т.е. единственные пересечения, которые допустимы — это пересечения i -го отрезка с $i + 1$ -м (а также n -го с первым) по одной и только одной точке.

Примеры

<code>broken.in</code>	<code>broken.out</code>
4	0 0
2 2	2 2
0 0	3 2
3 2	4 2
4 2	