

Задача А. Результаты олимпиады

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

N участников олимпиады получили уникальные номера от 1 до N . В результате решения задач на олимпиаде каждый участник получил некоторое количество баллов (целое число от 0 до 600). Известно, кто сколько баллов набрал.

Требуется перечислить участников олимпиады в порядке невозрастания набранных ими баллов.

Формат входного файла

В первой строке входного файла содержится число N ($1 \leq N \leq 100$). Далее записаны через пробел N чисел — количество набранных участниками баллов (1-е число — это баллы, набранные участником номер 1, 2-е — участником номер 2 и т.д.).

Формат выходного файла

В выходной файл следует вывести N чисел — номера участников в порядке невозрастания набранных ими баллов (номера участников, набравших одинаковое количество баллов, выводятся в порядке возрастания).

Примеры

<code>olymp.in</code>	<code>olymp.out</code>
5 100 312 0 312 500	5 2 4 1 3

Задача В. Сортировка подсчетом

Имя входного файла: `countsort.in`
 Имя выходного файла: `countsort.out`
 Ограничение по времени: 2 секунды
 Ограничение по памяти: 4 мегабайта

Вам дан массив. Требуется его отсортировать.

Формат входного файла

В первой строке входного файла находится одно целое число n — количество элементов массива ($1 \leq n \leq 200000$). Во второй строчке находятся n натуральных чисел — элементы массива. Все элементы массива не превосходят 10^4 .

Формат выходного файла

В единственную строку выходного файла выведите отсортированный массив.

Примеры

<code>countsort.in</code>	<code>countsort.out</code>
3 1 2 3	1 2 3
3 3 2 1	1 2 3

Note

Используйте силу, силу сортировки подсчетом.

Задача С. Ярый коллекционер бабочек

Имя входного файла: `collect.in`
 Имя выходного файла: `collect.out`
 Ограничение по времени: 4 секунды
 Ограничение по памяти: 64 мегабайта

Как известно, Андрей Сергеевич — ярый коллекционер бабочек. Он имеет огромную коллекцию, экспонаты которой собраны со всего мира. Будем считать, что в мире существует 2 000 000 000 видов бабочек.

Чтобы не запутаться, Андрей Сергеевич присвоил каждому виду уникальный номер. Нумерация видов бабочек начинается с единицы.

Теперь он хочет знать, есть ли бабочка с видом K в его коллекции, или же её придётся добывать, затрачивая уйму сил и денег.

Формат входного файла

В первой строке входного файла содержится единственное число N ($1 \leq N \leq 100\,000$) — количество видов бабочек в коллекции Андрея Сергеевича.

В следующей строке через пробел находятся N упорядоченных по возрастанию чисел — номера видов бабочек в коллекции.

Все виды бабочки в коллекции имеют различные номера.

В третьей строке файла записано число M ($1 \leq M \leq 100\,000$), количество видов бабочек, про которых Андрей Сергеевич хочет узнать, есть ли они у него в коллекции или же нет. В последней строке входного файла содержатся через пробел M чисел — номера видов бабочек, наличие которых необходимо проверить.

Формат выходного файла

Выходной файл должен содержать M строчек. Для каждого запроса выведите число "YES", если бабочка с данным номером содержится в коллекции, и "NO" — в противном случае.

Примеры

collect.in	collect.out
7	NO
10 47 50 63 89 90 99	NO
4	YES
84 33 10 82	NO

Задача D. Коровы - в стойла

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

На прямой расположены стойла, в которые необходимо расставить коров так, чтобы минимальное расстояние между коровами было как можно больше.

Формат входного файла

В первой строке вводятся числа N ($3 \leq N \leq 10\,000$) — количество стойл и K ($2 \leq K < N$) — количество коров. Во второй строке задаются N натуральных чисел в порядке возрастания — координаты стойл (координаты не превосходят 10^9).

Формат выходного файла

Выведите одно число — наибольшее возможное допустимое расстояние.

Примеры

cows.in	cows.out
5 3	99
1 2 3 100 1000	