

Задача А. Сортировка

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

Отсортируйте массив целых чисел в порядке неубывания, используя сортировку выбором минимума.

Формат входных данных

Первая строка входного файла содержит целое число N ($1 \leq N \leq 1000$), вторая строка — N целых чисел, по модулю не превышающих $2 \cdot 10^9$.

Формат выходных данных

Данные числа следует вывести в порядке неубывания.

Примеры

<code>sort.in</code>	<code>sort.out</code>
5 9 2 7 1 2	1 2 2 7 9

Замечание

При решении этой задачи нельзя пользоваться стандартными функциями и методами `min`, `index`, `sort`, `sorted` и т. д.

Естественно, можно пользоваться функциями `min`, `max`, которые принимают два числа.

Задача В. Ревизия

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

В связи с визитом Императора Палпатина было решено обновить состав дроидов в ангаре 32. Из-за кризиса было решено новых дроидов не закупать, но выкинуть пару старых. Как известно, Палпатин не переносит дроидов с маленькими серийными номерами, так что всё, что требуется — найти среди них двух, у которых серийные номера наименьшие.

Формат входных данных

Первая строка входного файла содержит целое число N — количество дроидов ($2 \leq N \leq 100\,000$), вторая строка — N целых чисел, по модулю не превышающих $2 \cdot 10^9$ — номера дроидов.

Формат выходных данных

Выведите два числа: первым — наименьший серийный номер дроида (которого поэтому следует утилизировать в первую очередь), а вторым — второй по минимальности.

Примеры

<code>inspection.in</code>	<code>inspection.out</code>
5 49 100 23 -100 157	-100 23
3 1 2419 1	1 1

Замечание

При решении этой задачи нельзя пользоваться стандартными функциями и методами `min`, `index`, `sort`, `sorted` и т. д.

Естественно, можно пользоваться функциями `min`, `max`, которые принимают два числа.

Задача С. Заменить цифру

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

Дано число N , и последовательность из N цифр, а также цифры A и B . Заменить в этой последовательности каждое вхождение цифры A на цифру B .

Формат входных данных

Программе дано число N , в следующей строке дано N цифр, в третьей строке цифры A, B ($1 \leq N \leq 10^7$).

Формат выходных данных

Вывести полученную последовательность цифр.

Примеры

<code>change.in</code>	<code>change.out</code>
4 2 1 4 1 2 5	5 1 4 1

Задача D. Гадалка

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

Вася пришел к гадалке, чтобы узнать, сколько добрых дел он должен сделать в этом году, чтобы год прошел удачно. Гадалка дала ему список из N чисел и сказала, что количество добрых дел, которые он должен сделать, соответствует одному из чисел этого списка, причем произведение цифр числа должно быть приближено к магическому числу K . Вася поверил гадалке и просит у Вас помощи. Вам даны числа из списка, который предоставила гадалка Васе, и число K . Найдите такое число из этого списка, произведение цифр которого максимально приближено к K . Если таких чисел несколько, выведите то, которое стоит раньше в списке.

Формат входных данных

В первой строке входного файла содержится два натуральных числа N ($N \leq 1000$) — количество чисел в списке, и K ($K \leq 10000$) — магическое число, во второй строке N натуральных чисел, каждое из которых в диапазоне от 1 до 10000.

Формат выходных данных

В выходной файл выведите нужное число из списка.

Примеры

<code>soot.in</code>	<code>soot.out</code>
5 16 11 8 29 6 20	29

Задача E. За мной просили не занимать

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

На одну ленту, вставляемую в кассовый аппарат, помещается L строк текста. Описание каждого приобретаемого товара занимает в чеке m строк, никакой дополнительной информации в чеке не печатается. Кассир меняет ленту либо в конце дня, либо когда видит, что покупки следующего покупателя не поместятся на ленте. Ваша задача — определить, после какого покупателя кассиру придется впервые за день поменять ленту в аппарате

Формат входных данных

В первой строке заданы два числа: длина ленты L ($1 \leq L \leq 1000$) и количество строк, которые занимает один товар ($1 \leq m \leq 50$).

Во второй строке задано число n — количество покупателей, пришедших в течение дня ($1 \leq n \leq 1000$).

В третьей — через пробел указано количество товаров, приобретаемых каждым покупателем. Гарантируется, что покупки каждого покупателя могут уместиться на одной ленте.

Формат выходных данных

Одно число — номер покупателя, после которого придется поменять ленту (покупатели нумеруются с 1).

Примеры

<code>receipts.in</code>	<code>receipts.out</code>
10 1 5 2 4 3 5 1	3
7 2 4 1 2 1 1	2

Замечание

Во втором тесте: первые два покупателя совершили в сумме 3 покупки и заняли на ленте 6 строк. Осталась одна свободная строка. Покупки третьего покупателя занимают 2 строки и не влезают на ленту, следовательно перед ним ленту надо менять.

Задача F. Результаты олимпиады

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

В олимпиаде участвовали n человек. Каждый получил определенное количество баллов, при этом оказалось, что у всех участников — разное число баллов. Упорядочите список участников олимпиады в порядке убывания набранных баллов.

Формат входных данных

В первой строке входных данных записано количество участников олимпиады n ($1 \leq n \leq 100$). Далее следует n строк, в каждой строке записана фамилия участника, затем, через пробел, набранное им количество баллов. Все фамилии — это различные последовательности прописных и строчных латинских букв длины от 1 до 20. Баллы — различные целые числа от 1 до 100.

Формат выходных данных

Выведите список участников (только фамилии) в порядке убывания набранных баллов. Фамилии выводите по одной в строке.

Примеры

<code>olymp.in</code>	<code>olymp.out</code>
3	Sidorov
Ivanov 15	Ivanov
Petrov 10	Petrov
Sidorov 20	

Задача G. Сортировка времени

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

Вводится сначала число N ($1 \leq N \leq 100$), а затем N моментов времени. Каждый момент времени задается 3 целыми числами - часы (от 0 до 23), минуты (от 0 до 59) и секунды (от 0 до 59), каждый из них записан в отдельной строке.

Формат входных данных

Выведите моменты времени, упорядоченные в порядке неубывания (момент времени также выводится в виде трех чисел, ведущие нули выводить не обязательно).

Формат выходных данных

Примеры

<code>time.in</code>	<code>time.out</code>
4	7 30 0
10	10 20 30
20	13 30 30
30	23 59 59
7	
30	
00	
23	
59	
59	
13	
30	
30	

Задача Н. Перепись населения

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

В главном корпусе живет N человек, возраст каждого варьируется от 0 до 1000 лет. Как-то раз с утра преподаватели решили выяснить, людей какого возраста в главном корпусе больше всего. Разумеется, выполнять эту работу придется вам. Найдите, какой возраст встречается чаще всего, и выведите его.

Формат входных данных

В первой строчке дано целое число N — количество обитателей главного корпуса ($1 \leq N \leq 10^5$). Во второй строчке даны N чисел — их возраста. Каждый возраст — целое число от 0 до 1000.

Формат выходных данных

Выведите самый часто встречающийся возраст (если таких несколько, выведите возраст самых младших).

Примеры

<code>census.in</code>	<code>census.out</code>
2 1 2	1
5 31 861 202 273 672	31
7 16 15 14 17 15 16 16	16