

## Задача 1. Сравнение элементов

Источник: базовая  
Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: разумное

Дан массив из  $N$  чисел:  $b_1, b_2, \dots, b_N$ .

Требуется для каждого элемента массива  $b_i$  посчитать количество элементов  $b_j$ , стоящих правее ( $i < j$ ), но меньших него ( $b_i > b_j$ ).

### Формат входных данных

В первой строке записано целое число  $N$  ( $1 \leq N \leq 100$ ).

Во второй строке через пробел записано  $N$  целых положительных чисел — значения элементов массива. Гарантируется, что значения элементов не превосходят 100.

### Формат выходных данных

Выведите  $N$  целых чисел через пробел: по порядку для каждого элемента входного массива выведите количество элементов, удовлетворяющих описанному выше условию.

### Примеры

input.txt	output.txt
5 4 3 5 1 2	3 2 2 0 0
3 3 2 1	2 1 0
4 1 2 3 3	0 0 0 0

### Пояснение к примеру

В первом примере:

- Правее числа 4 стоят числа 3, 5, 1 и 2, **три** из которых имеют значение меньше 4.
- Правее числа 3 стоят числа 5, 1 и 2, **два** из которых имеют значение меньше 3.
- Правее числа 5 стоят числа 1 и 2, **два** из которых имеют значение меньше 5.
- Для чисел 1 и 2 справа нет чисел, имеющих значение меньше.

## Задача 2. Суммы $k$ -ых

Источник: базовая  
Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: разумное

Дан массив из  $N$  чисел  $a_1, a_2, \dots, a_N$ .

Требуется вывести  $N$  чисел  $t_k$  ( $k = 1, \dots, N$ ), где  $t_k$  — сумма элементов массива с шагом  $k$ :

$$t_k = \sum_{j=1}^{N/k} a_{j \cdot k}$$

Т.е.  $t_1$  равняется сумме всех элементов массива,  $t_2$  равняется сумме каждого второго элемента массива,  $t_3$  равняется сумме каждого третьего элемента и т.д.

### Формат входных данных

В первой строке записано целое число  $N$  ( $1 \leq N \leq 10^5$ ).

Во второй строке через пробел записано  $N$  целых чисел  $a_i$  ( $1 \leq a_i \leq 10^4$ ).

### Формат выходных данных

Выведите  $N$  строк: в  $k$ -й строке выведите значение  $t_k$ .

### Пример

input.txt	output.txt
6	24
4 3 5 1 2 9	13
	14
	1
	2
	9

### Пояснение к примеру

$$t_1 = a_1 + a_2 + a_3 + a_4 + a_5 + a_6 = 4 + 3 + 5 + 1 + 2 + 9 = 24$$

$$t_2 = a_2 + a_4 + a_6 = 3 + 1 + 9 = 13$$

$$t_3 = a_3 + a_6 = 5 + 9 = 14$$

$$t_4 = a_4 = 1$$

$$t_5 = a_5 = 2$$

$$t_6 = a_6 = 9$$

## Задача 3. Гистограмма

Источник: базовая  
Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: разумное

Дан массив из  $N$  чисел  $a_0, a_1, \dots, a_{N-1}$ .

Постройте гистограмму значений элементов массива: для каждого значения подсчитайте сколько раз оно встречается в массиве.

### Формат входных данных

В первой строке входного файла записано целое число  $N$  ( $1 \leq N \leq 10^5$ ).

Во второй строке через пробел записано  $N$  целых чисел  $a_i$  ( $1 \leq a_i \leq 10^4$ ).

### Формат выходных данных

Для каждого значения, которое встречается в массиве хотя бы раз, в выходной файл требуется вывести сколько раз оно присутствует среди элементов массива в формате: “значение: количество”.

Значения требуется выводить в порядке возрастания.

Для вывода двух целых чисел, разделённых двоеточием с пробелом, удобно использовать функцию `printf` со следующей форматной строкой:

```
printf("%d: %d", value, count);
```

### Пример

input.txt	output.txt
10	1: 2
3 4 5 10 3 4 10 1 1 3	3: 3
	4: 2
	5: 1
	10: 2

## Задача 4. Различные числа

Источник:	основная
Имя входного файла:	<code>input.txt</code>
Имя выходного файла:	<code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	разумное

Задана последовательность целых чисел  $a_1, a_2, \dots, a_n$ . Сосчитать, сколько различных чисел в этой последовательности. Например, в массиве из четырех чисел 5,7,5,5 различных чисел два.

### Формат входных данных

В первой строке входного файла записано число  $n$  ( $1 \leq n \leq 100000$ ).

Во второй строке через пробел записаны целые числа  $a_1, a_2, \dots, a_n$ , каждое из которых по модулю не превосходит  $10^3$ .

### Формат выходных данных

В выходной файл необходимо вывести целое число – количество различных чисел.

### Примеры

<code>input.txt</code>	<code>output.txt</code>
10 1 2 3 5 4 1 2 2 1 4	5

## Задача 5. Решето Эратосфена

Источник:	основная
Имя входного файла:	<code>input.txt</code>
Имя выходного файла:	<code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	разумное

Задано натуральное число  $N$ . Необходимо найти все простые числа, не превосходящие заданного  $N$  и вывести их в порядке возрастания. Использовать алгоритм «Решето Эратосфена»:

Шаг 0. Образовать из целых чисел от 2 до  $N$  множество  $M$ . Для моделирования множества использовать массив. Выбрать в нем минимальный по значению элемент  $K$  (это 2).

Шаг 1. Удалить из множества все числа, большие  $K$ , которые делятся на  $K$  без остатка. Это все числа, отстоящие друг от друга на  $K$ , начиная с числа  $2 * K$ .

Шаг 2. Переменной  $K$  присвоить значение следующего минимального элемента из множества (это будет следующее простое число). Если  $K * K \leq N$ , то перейти на Шаг 1.

Шаг 3. Выдать значения всех элементов множества  $M$  в возрастающем порядке.

### Формат входных данных

Во входном файле записано одно натуральное число  $N$  ( $2 \leq N \leq 10^6$ ).

### Формат выходных данных

В выходной файл необходимо вывести в порядке возрастания через пробел все простые числа, не превосходящие  $N$ .

### Пример

<code>input.txt</code>	<code>output.txt</code>
23	2 3 5 7 11 13 17 19 23

## Задача 6. Сумма

Источник: основная  
Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: разумное

Дан массив из  $N$  чисел  $a_0, a_1, \dots, a_{N-1}$ .

Требуется найти два индекса  $L$  и  $R$  ( $L \leq R$ ) таких, что сумма  $a_L + a_{L+1} + \dots + a_{R-1} + a_R$  будет максимальной.

### Формат входных данных

В первой строке записано целое число  $N$  ( $1 \leq N \leq 5\,000$ ).

Во второй строке через пробел записано  $N$  целых чисел, каждое из которых по модулю не превосходит 10 000.

### Формат выходных данных

Выведите три целых числа через пробел:  $L$ ,  $R$  и сумму элементов с индексами от  $L$  до  $R$ .

Если существует несколько вариантов выбрать такие  $L$  и  $R$ , что сумма элементов будет максимальной, то требуется вывести вариант с минимальным  $L$ . Если существует несколько с минимальным  $L$ , то среди таких требуется выбрать вариант с минимальным  $R$ .

### Примеры

input.txt	output.txt
3 1 2 3	0 2 6
5 1 -2 3 4 5	2 4 12
4 2 -2 1 1	0 0 2

## Задача 7. Бинарный поиск

Источник:	основная
Имя входного файла:	<code>input.txt</code>
Имя выходного файла:	<code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	разумное

В упорядоченном целочисленном массиве методом бинарного поиска найти заданное число и вывести его номер.

### Формат входных данных

В первой строке входного файла записаны через пробел два целых числа  $N$  и  $X$  – размер массива и число, которое нужно найти в этом массиве ( $1 \leq N \leq 10^6$ ,  $-10^6 \leq X \leq 10^6$ ).

Во второй строке через пробел записаны  $N$  целых чисел в порядке неубывания – элементы массива. Все числа по модулю не превосходят  $10^6$ .

### Формат выходных данных

В выходной файл необходимо вывести номер элемента массива, содержащего заданное число  $X$ . Если такого числа нет, то вывести число  $-1$ .

### Пример

<code>input.txt</code>	<code>output.txt</code>
5 9 2 4 7 9 12	3

## Задача 8. Циклический сдвиг

Источник:	повышенной сложности
Имя входного файла:	<code>input.txt</code>
Имя выходного файла:	<code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	разумное

Задан массив целых чисел длины  $n$  и натуральное число  $k$ . Необходимо сдвинуть все элементы массива на  $k$  позиций вперед. Например, если  $k = 1$ , то элемент с позиции 0 переместится в позицию 1, с позиции 1 – в позицию 2, ... с позиции  $(n - 1)$  – в позицию 0.

### Формат входных данных

В первой строке входного файла записаны через пробел два числа  $n$  и  $k$  ( $1 \leq n \leq 10^5, 0 \leq k \leq 10^6$ ).

Во второй строке через пробел записаны целые числа  $a_0, a_1, a_2, \dots, a_{n-1}$ , каждое из которых по модулю не превосходит  $10^4$ .

### Формат выходных данных

В выходной файл необходимо вывести перестроенный массив. Числа выводить через пробел. После последнего числа в массиве поставить перевод строки.

### Пример

input.txt	output.txt
10 1 1 2 3 5 4 1 2 2 1 4	4 1 2 3 5 4 1 2 2 1



## Задача 9. Бинарный поиск 2

Источник: повышенной сложности  
Имя входного файла: `input.txt`  
Имя выходного файла: `output.txt`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: разумное

В упорядоченном массиве вещественных чисел методом бинарного поиска найти самое близкое к заданному и вывести его номер.

### Формат входных данных

В первой строке входного файла записаны через пробел два целых числа  $N$  и  $X$  – размер массива и число, близкое к которому нужно найти в этом массиве ( $1 \leq N \leq 10^6$ ,  $-10^6 \leq X \leq 10^6$ ).

Во второй строке через пробел записаны  $N$  вещественных чисел в порядке неубывания – элементы массива. Все числа по модулю не превосходят  $10^7$ .

### Формат выходных данных

В выходной файл необходимо вывести номер элемента массива, содержащего заданное число, значение которого наиболее близко к  $X$ . Если таких чисел больше одного, то вывести номер первого по порядку.

### Пример

input.txt	output.txt
5 5 2.1 4.37 6.2 9.07 12.01	1

## Задача 10. Даты

Источник:	повышенной сложности
Имя входного файла:	<code>input.txt</code>
Имя выходного файла:	<code>output.txt</code>
Ограничение по времени:	1 секунда
Ограничение по памяти:	разумное

По двум заданным датам нужно посчитать количество дней, прошедших между ними.

### Формат входных данных

Входной файл состоит из двух строк. Каждая строка содержит описание одной даты. Это три целых числа, записанные через пробел – день, месяц, год, соответственно.

Годы могут принимать значения от 1 до 2050, месяцы от 1 до 12, а дни – от 1 до максимального номера дня текущего месяца. Даты могут задаваться в любом порядке.

Следует помнить, что встречаются високосные годы. Год является високосным, если он делится нацело на 4 и не делится на 100, либо делится на 400. Например, 2000 год – високосный, а 1900 – нет.

### Формат выходных данных

Если обе даты заданы корректно, то в выходной файл необходимо вывести неотрицательное целое число – количество дней, прошедших между датами. В противном случае необходимо вывести слово `ERROR`.

### Примеры

<code>input.txt</code>	<code>output.txt</code>
1 1 2012 2 1 2012	1
29 2 2012 29 2 2011	ERROR