

## Задача А. В поисках минимума

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

В ЛКШ переполох. Потерялся минимум. Вы должны его найти.

### Формат входного файла

В первой строке — число  $N$  ( $1 \leq N \leq 900$ ). В строках со второй по  $N + 1$  находится по одному числу. Все числа не превышают по модулю  $10^5$ .

### Формат выходного файла

Выведите минимум среди этих  $N$  чисел.

### Примеры

<code>find-min.in</code>	<code>find-min.out</code>
4	-4
5	
8	
-4	
6	

### Примечание

Требуется написать функцию «`my_min`», которая будет принимать на вход список и возвращать значение минимума.

## Задача В. МегаНОД

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 64 мегабайта

Дано несколько чисел. Найти самое большое число, на которое делятся все несколько чисел.

### Формат входного файла

В единственной строке даны через пробел несколько чисел ( $1 \leq \text{несколько} \leq 1000$ ).

### Формат выходного файла

Выведите искомое число.

### Примеры

<code>stdin</code>	<code>stdout</code>
18 30 21	3

### Примечание

Ваша программа должна содержать две функции. Первая должна принимать на вход список и возвращать НОД всех чисел списка. Вторая должна вычислять НОД двух чисел.

## Задача С. Ханойские башни

Имя входного файла: `hanoi.in`  
Имя выходного файла: `hanoi.out`  
Ограничение по времени: 6 секунды  
Ограничение по памяти: 64 мегабайта

Даны три стержня. На первом стержне находятся несколько дисков сверху вниз по возрастанию размера диска. Два другие пустые. Требуется перенести все диски с первого стержня на второй. Переносить диски разрешается только по одному. Не разрешается класть больший диск на меньший. Для предотвращения заикливания, как это произошло с буддийскими монахами, не используйте циклы в программе.

### Формат входного файла

Вводится одно число  $n$  ( $1 \leq n \leq 19$ ) — количество дисков на первом стержне.

### Формат выходного файла

Выведите по два числа в строке — номера стержней, откуда и куда переносится диск. Решение должно быть кратчайшим.

### Примеры

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

## Задача D. Новый sum

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

Стандартная функция `sum` не умеет складывать строки. Поэтому, у создателей языка программирования `python` есть задание для талантливых учеников. Им (т.е. вам) поручается создать свою усовершенствованную функцию `sum`, которая умеет складывать как числа, так и строки.

### Формат входного файла

В первой строке входного файла находится много чисел ( $1 \leq \text{много} \leq 100000$ ), разделённых пробелом. Во второй строке файла — несколько строк ( $1 \leq \text{несколько} \leq 100$ ), также разделённых пробелом и длиной не более 100 символов.

### Формат выходного файла

В выходной файл выведите две строки: в первой — сумму чисел, во второй — сумму строк.

### Примеры

sum.in	sum.out
2 6 11 4	23
There are too much spaces	Therearetoomuchspaces

### Примечание

- стандартную функцию `sum` использовать запрещено
- не называйте функцию `sum`. Назовите её по-другому (например, `my_sum`).