

## Задача 1. Автобусные остановки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.5 секунд
Ограничение по памяти:	512 мегабайт

Автобусные остановки расположены через каждые  $K$  метров от начала улицы, то есть на расстоянии  $0, K, 2K, 3K$  и т.д. метров от начала. Света прошла от начала улицы  $N$  метров, после чего устала и захотела сесть на автобус. Определите, сколько метров нужно пройти Свете до ближайшей остановки.

### Формат входных данных

Программа получает на вход два целых числа  $K$  и  $N$ , записанных в отдельных строках.  $1 \leq K \leq 2 \times 10^9, 1 \leq N \leq 2 \times 10^9$ .

### Формат выходных данных

Программа должна вывести одно целое число — расстояние до ближайшей остановки.

### Система оценки

Решение, правильно работающее только для случаев, когда числа  $K$  и  $N$  не превосходят 10000, будет оцениваться в 60 баллов.

### Пример

стандартный ввод	стандартный вывод
600 2000	200

### Замечание

Пояснение к примеру. Остановки расположены на расстоянии  $0, 600, 1200, 1800$  и т.д. метров. Света прошла 2000 метров, до ближайшей остановки нужно идти 200 метров.

## Задача 2. Наборы пирожных

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	0.4 секунд
Ограничение по памяти:	512 мегабайт

На складе кондитерской фабрики хранятся пирожные двух видов — круассаны и эклеры. Круассанов  $A$  штук, а эклеров —  $B$  штук. Есть неограниченный запас подарочных коробок, в каждую коробку можно положить только три пирожных. При этом требуется, чтобы в коробке были пирожные обоих видов, то есть в одну коробку можно положить два круассана и один эклер или один круассан и два эклера.

Определите, можно ли упаковать все имеющиеся пирожные в коробки и выведите подходящий способ размещения пирожных по коробкам.

### Формат входных данных

Программа получает на вход два целых числа  $A$  и  $B$ , записанных в отдельных строках.  $1 \leq A \leq 10^9$ ,  $1 \leq B \leq 10^9$ .

### Формат выходных данных

Если можно разложить все пирожные по коробкам в соответствии с условием задачи, программа должна вывести два целых числа. Первое число равно количеству коробок, в которых лежит два круассана и один эклер. Второе число равно количеству коробок, в которых лежит один круассан и два эклера.

Если разложить все пирожные по коробкам нужным способом нельзя, программа должна вывести одно число  $-1$ .

### Система оценки

Решение, правильно работающее только для случаев, когда числа  $A$  и  $B$  не превосходят 100, будет оцениваться в 60 баллов.

### Примеры

стандартный ввод	стандартный вывод
4 5	1 2
5 3	-1

### Замечание

В первом примере нужно взять одну коробку с двумя круассанами и одним эклером и две коробки с одним круассаном и двумя эклерами. Всего получится 4 круассана и 5 эклеров.

### Задача 3. Мирные ладьи

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 512 мегабайт

На шахматной доске размером  $N \times N$  расставлено  $N$  шахматных ладей не бьющих друг друга, то есть на каждой вертикали и каждой горизонтали стоит ровно одна ладья.

	1	2	3	4	5
1				♖	
2		♗			
3			♘		
4					♙
5	♚				

Шахматную доску повернули на  $90^\circ$  по часовой стрелке. Выведите получившуюся расстановку ладей.

	1	2	3	4	5
1	♚				
2				♙	
3			♘		
4					♗
5		♖			

#### Формат входных данных

Первая строка входных данных содержит целое число  $N$ ,  $1 \leq N \leq 10^5$  — размер доски. Следующие  $N$  строк содержат по одному числу от 1 до  $N$ , а именно, в  $i$ -й строке записано число  $a_i$  — номер вертикали, в которой стоит ладья на  $i$ -й горизонтали. В этой задаче горизонтали нумеруются числами от 1 до  $N$  сверху вниз, вертикали нумеруются числами от 1 до  $N$  слева направо (см. рисунок).

#### Формат выходных данных

Программа должна вывести  $N$  чисел — расстановку ладей после поворота в таком же формате.

#### Пример

стандартный ввод	стандартный вывод
5	1
4	4
2	3
3	5
5	2
1	

#### Замечание

Пример в условии соответствует рисункам. Первоначально ладьи стояли в столбцах 4, 2, 3, 5, 1 при перечислении их по строкам сверху вниз. После поворота ладьи стоят в столбцах 1, 4, 3, 5, 2.

## Задача 4. Таблица

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 0.5 секунд  
Ограничение по памяти: 512 мегабайт

Бесконечную таблицу, строки и столбцы которой пронумерованы целыми числами начиная с 1 сверху вниз и слева направо, заполняют целыми числами 1, 2, 3 и т.д. Числа выписываются в соседние клетки по границам квадратов увеличивающегося размера (см. рисунок).

	1	2	3	4	5
1	1	2	9	10	
2	4	3	8	11	
3	5	6	7	12	
4	16	15	14	13	
5	17	18	...		

Дано число  $n$ , определите номер строки и номер столбца, в котором окажется это число.

### Формат входных данных

Программа получает на вход одно целое число  $n$ ,  $1 \leq n \leq 10^{18}$ .

Обратите внимание, что значение  $n$  может быть больше, чем возможное значение 32-битной целочисленной переменной, поэтому необходимо использовать 64-битные числа (тип `int64` в языке `Pascal`, тип `long long` в `C` и `C++`, тип `long` в `Java` и `C#`).

### Формат выходных данных

Программа должна вывести два целых числа: номер строки и номер столбца, в которых находится число  $n$  в этой таблице. Запись выводимых чисел должна содержать только цифры, вывод действительных чисел в ответе считается неверным.

### Система оценки

Решение, правильно работающее только для случаев, когда  $n \leq 100$ , будет оцениваться в 20 баллов.

Решение, правильно работающее только для случаев, когда  $n \leq 10^4$ , будет оцениваться в 40 баллов.

Решение, правильно работающее только для случаев, когда  $n \leq 10^9$ , будет оцениваться в 60 баллов.

### Пример

стандартный ввод	стандартный вывод
15	4 2

## Задача 5. Agar.io

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	512 мегабайт

В многопользовательской игре Agar.io игроки управляют бактериями. У каждой бактерии есть *размер* — целое положительное число. Если встречаются две бактерии разного размера, то бактерия большего размера поглощает меньшую бактерию. При этом меньшая бактерия исчезает, а размер большей бактерии увеличивается на размер меньшей бактерии. Если встречаются две бактерии равного размера, то ничего не происходит. Побеждает игрок, чья бактерия останется на игровом поле одна.

В игре участвуют  $n$  игроков, вам даны размеры их бактерий. Определите, какие из игроков имеют возможность выиграть в этой игре.

### Формат входных данных

Программа получает на вход целое число  $n$ ,  $1 \leq n \leq 10^5$  — количество игроков. Следующие  $n$  строк содержат по одному числу  $a_i$  — размеры бактерий,  $1 \leq a_i \leq 10^9$ . Числа  $a_i$  заданы в порядке неубывания.

### Формат выходных данных

Программа должна вывести  $n$  чисел равных «0» или «1», по одному числу в строке. Если  $i$ -е число равно 0, то это означает, что  $i$ -й игрок (размер бактерии которого первоначально был равен  $a_i$ ) ни при каких обстоятельствах не может выиграть в этой игре. Если  $i$ -е число равно 1, то это означает, что  $i$ -й игрок имеет возможность выиграть в этой игре.

### Система оценки

Решение, правильно работающее только для случаев, когда  $n \leq 100$  и все  $a_i \leq 10^6$ , будет оцениваться в 60 баллов.

### Пример

стандартный ввод	стандартный вывод
4	0
1	0
1	1
3	1
4	

### Замечание

В примере из условия 4 бактерии размерами 1, 1, 3, 4. Бактерии размером 1 никого не могут съесть, поэтому не могут выиграть. Бактерия размером 4 может съесть всех. Бактерия размером 3 может съесть по очереди две бактерии размером 1. Тогда её размер станет 5, после этого она сможет съесть бактерию размером 4 и выиграть. Ответ: 0, 0, 1, 1.