

## Задача D. $(p, q)$ -лошадь

Входной файл: horse.in  
Выходной файл: horse.out  
Ограничение по времени: 1 секунда  
Ограничение памяти: 64 мегабайта

$(p, q)$ -лошадь представляет собой обобщение шахматной фигуры «конь». Она ходит на  $p$  клеток в одном направлении и  $q$  в перпендикулярном. Обычный шахматный конь, таким образом, является  $(2, 1)$ -лошадью.

Ваша задача — определить, сколько ходов требуется  $(p, q)$ -лошади, чтобы попасть из одной клетки доски размера  $M \times N$  в другую.

### Описание входного файла

Первая и единственная строка входного файла содержит целые числа  $M, N, p, q, x_1, y_1, x_2$  и  $y_2$  ( $1 \leq x_1, x_2 \leq M, 1 \leq y_1, y_2 \leq N, 0 \leq p \leq M \leq 100, 0 \leq q \leq N \leq 100$ ).

### Описание выходного файла

Первая строка выходного файла должна содержать целое число  $k$  — минимальное количество ходов, которое требуется  $(p, q)$ -лошади, чтобы попасть из клетки  $(x_1, y_1)$  в клетку  $(x_2, y_2)$ . Затем должны следовать  $k + 1$  строк, содержащие последовательные положения  $(p, q)$ -лошади на ее пути.

Если  $(p, q)$ -лошадь не может достичь клетки  $(x_2, y_2)$  из  $(x_1, y_1)$ , выведите  $-1$ .

### Пример

horse.in	horse.out
3 3 1 1 1 1 3 3	2 1 1 2 2 3 3
2 2 1 1 1 1 1 2	-1

## Задача E. Наименьшее общее кратное

Входной файл: lcm.in  
Выходной файл: lcm.out  
Ограничение по времени: 1 секунда  
Ограничение памяти: 64 мегабайта

Вычислите наименьшее общее кратное всех целых чисел от 1 до  $n$ .

### Описание входного файла

Входной файл содержит число  $n$  ( $1 \leq n \leq 1000$ ).

### Описание выходного файла

Выведите целое число — ответ на задачу.

### Пример

lcm.in	lcm.out
3	6

## Задача F. Наилучшее приближение

Входной файл: `nearest.in`  
Выходной файл: `nearest.out`  
Ограничение по времени: 4 секунды  
Ограничение памяти: 64 мегабайта

Вам задано  $N$  целых чисел. Ваша задача — расставить между последовательными числами знаки «+» и «-» таким образом, чтобы значение получившегося выражения было как можно ближе к заданному целому числу  $A$ .

### Описание входного файла

Первая строка входного файла содержит два целых числа:  $N$  ( $1 \leq N \leq 10000$ ) и  $A$  ( $A$  не превосходит 10000 по абсолютной величине). Следующие  $N$  строк содержат по одному целому числу  $X_i$  каждая (все числа не превосходят 10000 по абсолютной величине). Также гарантируется, что сумма абсолютных величин всех  $N$  чисел не превосходит 10000.

### Описание выходного файла

В первой строке выходного файла выведите значение получившегося выражения (которое должно быть как можно ближе к  $A$ ). Во второй строке выведите оптимальное выражение в форме  $X_1[+|-]X_2[+|-]\dots X_{N-1}[+|-]X_N$ .

### Пример

<code>nearest.in</code>	<code>nearest.out</code>
3 0	0
3	3+-2-1
-2	
1	

## Задача G. Строка

Входной файл: `string.in`  
Выходной файл: `string.out`  
Ограничение по времени: 1 секунда  
Ограничение памяти: 64 мегабайта

Для заданной строки  $S$  найдите ее порядковый номер в лексикографическом порядке всех строк, составленных из того же набора букв. Например, строка “abba” третья среди строк “aabb”, “abab”, “abba”, “baab”, “baba” и “bbaa”.

### Описание входного файла

Единственная строка входного файла содержит строку  $S$ , содержащую не более 32 маленьких латинских букв.

### Описание выходного файла

Выведите искомый номер.

### Пример

<code>string.in</code>	<code>string.out</code>
abba	3
baba	5