

**Задача А. Наибольшая возрастающая подпоследовательность**

Имя входного файла: `lis.in`  
Имя выходного файла: `lis.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Дана последовательность, требуется найти длину её наибольшей возрастающей подпоследовательности.

**Формат входного файла**

В первой строке входного файла задано число  $N$  — длина последовательности ( $1 \leq N \leq 1000$ ). Во второй строке задаётся сама последовательность (разделитель — пробел). Элементы последовательности — целые числа, не превосходящие 10 000 по модулю.

**Формат выходного файла**

Требуется вывести длину наибольшей возрастающей подпоследовательности.

**Примеры**

<code>lis.in</code>	<code>lis.out</code>
6 3 29 5 5 28 6	3

**Задача В. Наибольшая общая подпоследовательность**

Имя входного файла: `lcs.in`  
Имя выходного файла: `lcs.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Даны две последовательности. Найдите длину их наибольшей общей подпоследовательности (подпоследовательность — это то, что можно получить из данной последовательности вычеркиванием некоторых элементов).

**Формат входного файла**

В первой строке входного файла записано число  $N$  — длина первой последовательности ( $1 \leq N \leq 1000$ ). Во второй строке записаны члены первой последовательности (через пробел) — целые числа, не превосходящие 10 000 по модулю. В третьей строке записано число  $M$  — длина второй последовательности ( $1 \leq M \leq 1000$ ). В четвертой строке записаны члены второй последовательности (через пробел) — целые числа, не превосходящие 10 000 по модулю.

**Формат выходного файла**

В выходной файл требуется вывести единственное целое число: длину наибольшей общей подпоследовательности, или число 0, если такой не существует.

**Примеры**

<code>lcs.in</code>	<code>lcs.out</code>
3 1 2 3 4 2 1 3 5	2

**Задача С. Наибольшая последовательнократная подпоследовательность**

Имя входного файла: `sequence.in`  
Имя выходного файла: `sequence.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 64 мегабайта

Для заданной числовой последовательности  $a_1, a_2, \dots, a_n$  требуется найти длину максимальной последовательнократной подпоследовательности.

Для последовательнократной подпоследовательности  $a_{k_1}, a_{k_2}, \dots, a_{k_t}$  ( $k_1 < k_2 < \dots < k_t$ ) верно, что  $a_{k_i} | a_{k_j}$  при  $1 \leq i < j \leq t$  (утверждение « $a|b$ » эквивалентно « $b$  кратно  $a$ »). Подпоследовательность из одного элемента полагается последовательнократной по определению.

**Формат входного файла**

В первой строке входного файла записано одно натуральное число  $N$  ( $1 \leq N \leq 1000$ ) — количество чисел в исходной последовательности. Далее следует  $N$  натуральных чисел, не превосходящих  $2 \cdot 10^9$  — сама последовательность.

**Формат выходного файла**

Вывести единственное число, равное искомому количеству.

**Примеры**

<code>sequence.in</code>	<code>sequence.out</code>
4 3 6 5 12	3