

## Задача А. Количество ПСП

Имя входного файла: `quant.in`  
Имя выходного файла: `quant.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайт

Посчитайте количество правильных скобочных последовательностей длины  $2n$  ( $n$  открывающих скобок и  $n$  закрывающих), составленных из круглых и квадратных скобок так, что внутри любой пары круглых скобок нет квадратных скобок.

### Формат входных данных

В единственной строке через пробел записано целое неотрицательное число  $n$ , не превосходящее 1000.

### Формат выходных данных

Выведите остаток от деления количества искомых правильных скобочных последовательностей на  $10^9 + 7$ .

### Примеры

	<code>quant.in</code>	<code>quant.out</code>
1	1	2
2	2	7

## Задача В. Наибольшая общая подпоследовательность

Имя входного файла: `stdin`  
Имя выходного файла: `stdout`  
Ограничение по времени: 3 секунды  
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 входных данных

В первой строке входных данных записано целое число  $N$  — длина первой последовательности ( $1 \leq N \leq 1000$ ). Во второй строке записаны  $N$  чисел — члены первой последовательности. В третьей строке записано целое число  $M$  — длина второй последовательности ( $1 \leq M \leq 1000$ ). В четвертой строке записаны  $M$  чисел — члены второй последовательности. Члены последовательностей — целые числа, не превосходящие 10 000 по модулю.

### Формат выходных данных

Программа должна вывести единственное целое число — длину наибольшей общей подпоследовательности, или число 0, если такой не существует.

### Примеры

<code>stdin</code>	<code>stdout</code>
3 1 2 3 4 2 1 3 5	2

## Задача С. Невозрастающая подпоследовательность

Имя входного файла: `subseq.in`  
Имя выходного файла: `subseq.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Вам требуется написать программу, которая по заданной последовательности находит максимальную невозрастающую её подпоследовательность (т.е. такую последовательность чисел  $a_{i_1}, a_{i_2}, \dots, a_{i_k}$  ( $i_1 < i_2 < \dots < i_k$ ), что  $a_{i_1} \geq a_{i_2} \geq \dots \geq a_{i_k}$  и не существует последовательности с теми же свойствами длиной  $k + 1$ ).

### Формат входных данных

В первой строке задано число  $n$  — количество элементов последовательности ( $1 \leq n \leq 239\,017$ ). В последующих строках идут сами числа последовательности  $a_i$ , отделенные друг от друга произвольным количеством пробелов и переводов строки (все числа не превосходят по модулю  $2^{31} - 2$ ).

### Формат выходных данных

Вам необходимо выдать в первой строке выходного файла число  $k$  — длину максимальной невозрастающей подпоследовательности. Во второй строке должны быть выведены все номера элементов исходной последовательности  $i_j$ , образующих искомую подпоследовательность. Номера выводятся в порядке возрастания. Если оптимальных решений несколько, разрешается выводить любое.

### Примеры

<code>subseq.in</code>	<code>subseq.out</code>
5	3
5 8 10 4 1	1 4 5

## Задача D. Разбиение на слагаемые

Имя входного файла: `part.in`  
Имя выходного файла: `part.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Дано число  $N$ . Ваша задача — посчитать количество способов разбить число  $N$  на слагаемые так, чтобы все слагаемые были различны между собой.

### Формат входных данных

В единственной строке входных данных содержится число  $N$  ( $1 \leq N \leq 2000$ ).

### Формат выходных данных

Выведите искомое количество способов разложить  $N$  на слагаемые. Так как ответ может быть очень большим, выведите его по модулю 1 000 000 007.

### Примеры

<code>part.in</code>	<code>part.out</code>
7	5

## Задача Е. Разбиения на $K$ слагаемых

Имя входного файла: стандартный ввод  
Имя выходного файла: стандартный вывод  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 128 мегабайт

Для данных натуральных чисел  $N$  и  $K$  определите количество способов представить число  $N$  в виде суммы  $K$  натуральных слагаемых, если способы, отличающиеся только порядком слагаемых считать одинаковыми.

### Формат входных данных

Программа получает на вход два натуральных числа  $N$  и  $K$ , не превосходящих 400. Гарантируется, что ответ не превосходит  $2^{64} - 1$ .

### Формат выходных данных

Выведите ответ на задачу.

### Примеры

стандартный ввод	стандартный вывод
6 3	3
5 2	2

## Задача F. Кино

Имя входного файла: `cinema.in`  
Имя выходного файла: `cinema.out`  
Ограничение по времени: 1 секунда  
Ограничение по памяти: 256 мегабайта

Трое друзей летом посмотрели несколько фильмов. Для каждого школьника известно, какие фильмы и в каком порядке он посмотрел (естественно, если фильм кому-то из ребят сильно понравился, он мог его пересмотреть несколько раз). Так как друзья любят смотреть фильмы вместе, втроем они посмотрели максимальное возможное количество. Сколько же раз они встречались вместе?

### Формат входных данных

В первой строке входного файла заданы три числа:  $N$ ,  $M$  и  $K$  — количества фильмов, просмотренных каждым из друзей ( $1 \leq N, M, K \leq 300$ ). В следующих трех строках выписаны номера фильмов, просмотренных ими. У друзей огромные планы на будущее, поэтому фильмы нумеруются числами между 1 и  $10^9$ .

### Формат выходных данных

В первой строке выведите единственное число — максимальное кол-во просмотренных фильмов. В следующей строке выведите через пробел номера просмотренных фильмов.

### Примеры

<code>cinema.in</code>	<code>cinema.out</code>
3 3 4	2
1 2 3	1 3
1 3 10	
3 1 10 3	

## Задача G. Восстановление

Имя входного файла: `recover.in`  
Имя выходного файла: `recover.out`  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Дарья обнаружила ошибку в своей программе, которая удаляет все символы из строки кроме “(” и “)”. Оказывается, некоторые символы заменяются на что-то нечитаемое.

Теперь её заинтересовал вопрос, сколько различных правильных скобочных последовательностей длины  $2n$  могут являться результатом исправленного алгоритма, то есть не будут противоречить данным, которые она таки не потеряла.

### Формат входных данных

Единственная строка входного файла содержит строку из круглых скобок и знаков вопроса, где вопросами обозначены утраченные символы. Длина строки не превосходит 10 000.

### Формат выходных данных

Выведите одно число — количество различных скобочных последовательностей, удовлетворяющих шаблону Дарьи, по модулю  $10^9 + 7$ .

### Примеры

<code>recover.in</code>	<code>recover.out</code>
<code>(??())?</code>	2

### Замечание

Вместо `python3` при сдаче этой задачи используйте `pyru3`.

## Задача Н. Число возрастающих подпоследовательностей

Имя входного файла: subseq.in  
Имя выходного файла: subseq.out  
Ограничение по времени: 2 секунды  
Ограничение по памяти: 256 мегабайт

Задана последовательность из  $n$  чисел  $a_1, a_2, \dots, a_n$ . Подпоследовательностью длины  $k$  этой последовательности называется набор индексов  $i_1, i_2, \dots, i_k$ , удовлетворяющий неравенствам  $1 \leq i_1 < i_2 < \dots < i_k \leq n$ . Подпоследовательность называется возрастающей, если выполняются неравенства  $a_{i_1} < a_{i_2} < \dots < a_{i_k}$ .

Необходимо найти число возрастающих подпоследовательностей наибольшей длины заданной последовательности  $a_1, \dots, a_n$ . Так как это число может быть достаточно большим, необходимо найти остаток от его деления на  $10^9 + 7$ .

### Формат входных данных

Первая строка входного файла содержит целое число  $n$  ( $1 \leq n \leq 1000$ ). Вторая строка входного файла содержит  $n$  целых чисел:  $a_1, a_2, \dots, a_n$ . Все  $a_i$  не превосходят  $10^9$  по абсолютной величине.

### Формат выходных данных

В выходной файл выведите ответ на задачу.

### Примеры

subseq.in	subseq.out
5 1 2 3 4 5	1
6 1 1 2 2 3 3	8