

Задача А. Миллиардная Функция Васи

Имя входного файла: стандартный ввод
Имя выходного файла: стандартный вывод
Ограничение по времени: 2 секунды
Ограничение по памяти: 256 мегабайт

Вася — начинающий математик — решил сделать вклад в развитие этой науки и прославиться на весь мир. Но как это сделать, когда самые интересные факты, типа теоремы Пифагора, давно уже доказаны? Правильно! Придумать что-то свое, оригинальное. Вот юный математик и придумал Теорию Функций Васи, посвященную изучению поведения этих самых функций. Функции Васи (ФВ) устроены довольно просто: значением N -й ФВ в точке S будет количество чисел от 1 до N , имеющих сумму цифр S . Вам, как крутым программистам, Вася поручил найти значения миллиардной ФВ (то есть ФВ с $N = 10^9$), так как сам он с такой задачей не справится. А Вам слабо?

Формат входных данных

В единственной строке записано целое число S ($1 \leq S \leq 81$).

Формат выходных данных

Выведите одно число — значение миллиардной Функции Васи в точке S .

Примеры

стандартный ввод	стандартный вывод
1	10

Задача В. Интересные числа

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	256 мегабайт

Софья считает число интересным, если его цифры идут в неубывающем порядке. Например, числа 123, 1111 или 888999 – интересные.

Софья заинтересовалась, сколько существует интересных положительных чисел, лежащих в диапазоне от L до R включительно. Это число может оказаться довольно большим для больших L и R , поэтому Софья хочет найти остаток от деления этого числа на $10^9 + 7$.

Требуется написать программу, которая по заданным L и R определяет количество интересных чисел, лежащих в диапазоне от L до R включительно, и выводит остаток от деления этого числа на $10^9 + 7$.

Формат входных данных

Входной файл содержит две строки. Первая строка содержит число L , вторая строка содержит число R ($1 \leq L \leq R \leq 10^{100}$).

Формат выходных данных

Выходной файл должен одно целое число — остаток от деления количества интересных чисел, лежащих в диапазоне от L до R включительно, на $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
1 100	54

Задача С. Интервальные тренировки

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2.5 секунд
Ограничение по памяти:	512 мегабайт

В академии физической культуры разработали новый метод интервальных тренировок спортсменов. В соответствии с этим методом спортсмен должен тренироваться каждый день, однако рост нагрузки должен постоянно сменяться её снижением и наоборот.

План тренировки представляет собой набор целых положительных чисел a_1, a_2, \dots, a_m , где a_i описывает нагрузку спортсмена в i -й день. Любые два соседних дня должны иметь различную нагрузку: $a_i \neq a_{i+1}$. Чтобы рост нагрузки и её снижение чередовались, для i от 1 до $m - 2$ должно выполняться следующее условие: если $a_i < a_{i+1}$, то $a_{i+1} > a_{i+2}$, если же $a_i > a_{i+1}$, то $a_{i+1} < a_{i+2}$.

Суммарная нагрузка в процессе выполнения плана должна составлять n , то есть $a_1 + a_2 + \dots + a_m = n$. Ограничения на количество дней в плане нет, m может быть любым, но нагрузка в первый день тренировок зафиксирована: $a_1 = k$.

Прежде чем приступить к тестированию нового метода, руководство академии хочет выяснить, сколько различных планов тренировок удовлетворяет описанным ограничениям.

Требуется написать программу, которая по заданным n и k определяет, сколько различных планов тренировок удовлетворяют описанным ограничениям, и выводит остаток от деления количества таких планов на число $10^9 + 7$.

Формат входных данных

В первой строке входных данных находятся целые числа n, k ($1 \leq n \leq 5000, 1 \leq k \leq n$).

Формат выходных данных

Выведите одно число: остаток от деления количества планов тренировок на $10^9 + 7$.

Примеры

стандартный ввод	стандартный вывод
6 2	4
3 3	1

Замечание

В первом примере подходят следующие планы: $[2, 1, 2, 1]$, $[2, 1, 3]$, $[2, 3, 1]$, $[2, 4]$.

Во втором примере единственный подходящий план $[3]$.

Задача D. 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 Е. Лесенки

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

Лесенкой называется набор кубиков, в котором каждый горизонтальный слой содержит меньше кубиков, чем слой под ним (при этом слой начинается от левого края и не имеет дырок). Подсчитайте количество различных лесенок, которые могут быть построены из N кубиков.

Формат входных данных

Вводится одно число N ($1 \leq N \leq 150$).

Формат выходных данных

Выведите искомое количество лесенок.

Примеры

	<code>stairs.in</code>	<code>stairs.out</code>
	3	2

Задача F. 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 G. Количество ПСП

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

Посчитайте количество правильных скобочных последовательностей длины $2n$ (n открывающихся скобок и n закрывающихся), 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 Н. Неглубокие последовательности

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

Полина написала программу, удаляющую из строки все символы кроме "(" и ")". Теперь её заинтересовал вопрос, сколько различных правильных скобочных последовательностей длины $2n$ она может получить.

Известно, что Полина по политическим убеждениям запускает свою программу только на корректных математических выражениях, максимальная вложенность скобок в которых составляет в точности k .

Формат входных данных

Единственная строка входного файла содержит два числа n ($1 \leq n \leq 50$) и k ($1 \leq k \leq n$).

Формат выходных данных

Выведите одно число — искомое количество последовательностей по модулю $10^9 + 7$.

Примеры

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

Задача I. Гладиолус

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

Числовая последовательность задана рекуррентной формулой: $a_{i+1} = (ka_i + b) \bmod m$. Найдите её наибольшую возрастающую подпоследовательность.

Формат входных данных

Программа получает на вход пять целых чисел: длину последовательности n ($1 \leq n \leq 10^5$), начальный элемент последовательности a_1 , параметры k , b , m для вычисления последующих членов последовательности ($1 \leq m \leq 10^4$, $0 \leq k < m$, $0 \leq b < m$, $0 \leq a_1 < m$).

Формат выходных данных

Требуется вывести наибольшую возрастающую подпоследовательность данной последовательности, разделяя числа пробелами. Если таких последовательностей несколько, необходимо вывести одну (любую) из них.

Примеры

<code>gladiolus.in</code>	<code>gladiolus.out</code>
5 41 2 1 100	41 67 71

Замечание

В данном примере последовательность состоит из 5 элементов: $a_1 = 41$, $a_{i+1} = (2a_i + 1) \bmod 100$, то есть последовательность имеет вид 41, 83, 67, 35, 71.

Задача J. 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

Задача К. Painting Plan

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	2 секунды
Ограничение по памяти:	1024 мегабайта

В городе Иннополис продолжается строительство и модернизация, недавно в городе начали внедрение системы автоматической покраски заборов.

В качестве исходных данных системе дали набор отрезков $[l_i, r_i]$, которые должны быть покрашены, при этом отрезки могли пересекаться. Система проанализировала данные, и посчитала, что всего необходимо будет покрасить k метров забора (если участок забора входит в несколько отрезков, его все равно нужно покрасить только один раз).

К сожалению, из-за ошибки в программе, во время анализа данных, был испорчен файл с исходными данными. А именно, система взяла все числа l_i и r_i , положила их в общий массив x_i , и отсортировала его по возрастанию, утратив исходный порядок, а так же информацию о том, какое число является левой, а какое правой границей отрезка.

Вам нужно попытаться восстановить исходные отрезки по числу k и набору чисел x_i , либо сказать, что система дала сбой и такое невозможно.

Формат входных данных

В первой строке даны два целых числа n и k ($1 \leq n \leq 7000$; $0 \leq k \leq 30\,000$) — количество отрезков и общая длина покрашенной части забора, соответственно.

Во второй строке даны $2 \cdot n$ целых чисел x_1, x_2, \dots, x_{2n} ($0 \leq x_i \leq 30\,000$) — координаты концов отрезков в отсортированном порядке. Гарантируется, что все координаты различны.

Формат выходных данных

Если искомого плана не существует, то в первой строке выведите слово «No».

Иначе в первой строке выведите слово «Yes».

В следующих n строках выведите по два числа — индексы в массиве x_i , соответствующие левой и правой границе очередного отрезка. Индексы нумеруются с 1.

Если существует несколько правильных ответов, то выведите любой.

Примеры

стандартный ввод	стандартный вывод
4 9 0 1 3 5 8 9 10 12	Yes 4 5 1 2 3 6 7 8
3 2 1 2 3 4 5 6	No