

Задача А. Дождь

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

Город Ливнеполис представляет из себя поле длиной n и шириной m . Недавно в Ливнеполисе прошел дождь, и теперь в клетке (i, j) есть лужа глубиной $d_{i,j}$. В клетке $(1, 1)$ расположен Институт Ливнеполиса, а в клетке (n, m) — магазин, и именно по этому маршруту вам нужно пройти. Вы можете ходить из клетки (x, y) в клетки $(x + 1, y)$, $(x, y - 1)$ и $(x, y + 1)$ (если, конечно, такие клетки существуют). Вам нужно построить такой маршрут, чтобы на вашем пути максимальная глубина лужи была как можно меньше.

Формат входных данных

В первой строке даны n и m ($2 \leq n, m \leq 500$) — размеры Ливнеполиса. В следующих n строках даны по m чисел $d_{i,j}$ ($0 \leq d_{i,j} \leq 10^9$, $d_{1,1} = d_{n,m} = 0$).

Формат выходных данных

Выведите максимальную глубину лужи на найденном вами пути.

Примеры

rain.in	rain.out
3 3 0 1 9 9 1 9 9 1 0	1
5 3 0 1 1 9 9 2 1 1 1 2 9 9 1 1 0	2

Задача В. Кузнечик-2

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

У одного из преподавателей параллели С в комнате живёт кузнечик, который очень любит прыгать по клетчатой одномерной доске. Длина доски — N клеток. К его сожалению он умеет прыгать только на $1, 2, \dots, k$ клеток вперёд. А ещё кузнечик не может прыгать на одинаковое количество клеток два хода подряд.

Однажды преподавателям стало интересно, сколькими способами кузнечик может допрыгать из первой клетки до последней. Помогите им ответить на этот вопрос.

Формат входных данных

В первой и единственной строке входного файла записано два целых числа — N и k ($1 \leq N \leq 1000, 2 \leq k \leq 1000$).

Формат выходных данных

Выведите одно число — количество способов, которыми кузнечик может допрыгать из первой клетки до последней. Так как ответ может оказаться слишком большим, выведите его по модулю $10^9 + 7$.

Примеры

<code>grid-2.in</code>	<code>grid-2.out</code>
2 2	1
3 3	3

Задача С. Кино

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

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.

Пример

<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 D. Наибольшая возрастающая подпоследовательность

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

Дана последовательность, требуется найти её наибольшую возрастающую подпоследовательность.

Формат входных данных

В первой строке входных данных задано целое число n — длина последовательности ($1 \leq n \leq 5000$). Во второй строке задается сама последовательность. Числа разделяются пробелом. Элементы последовательности — целые числа, не превосходящие 10^9 по модулю.

Формат выходных данных

В первой строке выведите длину наибольшей возрастающей подпоследовательности, а во второй строке выведите через пробел саму наибольшую возрастающую подпоследовательность данной последовательности. Если ответов несколько — выведите любой.

Пример

стандартный ввод	стандартный вывод
6 3 29 5 5 28 6	3 3 5 28

Задача Е. Фрукты

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

Беси забралась на кухню Фермера Джона и обнаружила там кучу лимонов и апельсинов там (неограниченное количество и того и другого) и хочет съесть как можно больше. Максимум сытости Беси равен T ($1 \leq T \leq 5 \cdot 10^4$). Поедание апельсина увеличивает её сытость на A , а поедание лимона увеличивает её сытость на B ($1 \leq A, B \leq T$). Дополнительно, если она хочет, Беси может попить воды не более одного раза, что мгновенно уменьшит её сытость вдвое (с округлением вниз).

Помогите определить Беси максимальную сытость, которую она сможет достичь.

Формат входных данных

Первая и единственная строка содержит три целых числа T , A и B .

Формат выходных данных

Одно целое число, представляющее максимальную сытость, которую может достичь Беси.

Пример

<code>feast.in</code>	<code>feast.out</code>
8 5 6	8

Задача F. Ход конем

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

Шахматная ассоциация решила оснастить всех своих сотрудников такими телефонными номерами, которые бы набирались на кнопочном телефоне ходом коня. Например, ходом коня набирается телефон 340-49-27. При этом телефонный номер не может начинаться ни с цифры 0, ни с цифры 8.

1	2	3
4	5	6
7	8	9
	0	

Напишите программу, определяющую количество телефонных номеров длины N , набираемых ходом коня. Поскольку таких номеров может быть очень много, выведите ответ по модулю 10^9 .

Формат входных данных

Во входном файле записано целое число N ($1 \leq N \leq 100$).

Формат выходных данных

Выведите в выходной файл искомое количество телефонных номеров по модулю 10^9 .

Примеры

<code>phones.in</code>	<code>phones.out</code>
1	8
2	16

Задача G. Наибольшая общая подпоследовательность

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

Даны две последовательности, требуется найти и вывести их наибольшую общую подпоследовательность.

Формат входных данных

В первой строке входных данных содержится целое число n — длина первой последовательности ($1 \leq n \leq 2000$). Во второй строке заданы члены первой последовательности (через пробел) — целые числа, не превосходящие 10^9 по модулю. В третьей строке записано целое число m — длина второй последовательности ($1 \leq m \leq 2000$). В четвертой строке задаются члены второй последовательности (через пробел) — целые числа, не превосходящие 10^9 по модулю.

Формат выходных данных

В первой строке выведите длину наибольшей общей подпоследовательности, а во второй строке выведите через пробел саму наибольшую общую подпоследовательность данных последовательностей. Если ответов несколько — выведите любой.

Пример

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

Задача Н. Невозрастающая подпоследовательность

Имя входного файла: `subseq.in`
Имя выходного файла: `subseq.out`
Ограничение по времени: 1 секунда
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. В последующих строках должны быть выведены (по одному числу в каждой строке) все номера элементов исходной последовательности i_j , образующих искомую подпоследовательность. Номера выводятся в порядке возрастания. Если оптимальных решений несколько, разрешается выводить любое.

Пример

subseq.in	subseq.out
5	3
5	3 4 5
8	
10	
4	
1	

Задача I. Рюкзак

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

Дано n предметов массой m_1, \dots, m_n и стоимостью c_1, \dots, c_n соответственно.

Ими наполняют рюкзак, который выдерживает вес не более m . Определите набор предметов, который можно унести в рюкзаке, имеющий наибольшую стоимость.

Формат входных данных

В первой строке вводится натуральное число n , не превышающее 1000 и натуральное число m , не превышающее 10000.

Во второй строке вводятся n натуральных чисел m_i , не превышающих 100.

Во третьей строке вводятся n натуральных чисел c_i , не превышающих 100.

Формат выходных данных

В первой строке выведите количество предметов, которые нужно взять. Во второй строке выведите номера предметов (числа от 1 до n), которые войдут в рюкзак наибольшей стоимости.

Пример

стандартный ввод	стандартный вывод
4 6 2 4 1 2 7 2 5 1	3 1 3 4

Задача J. Банкомат

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

В некотором государстве в обращении находятся банкноты определенных номиналов. Национальный банк хочет, чтобы банкомат выдавал любую запрошенную сумму при помощи минимального числа банкнот, считая, что запас банкнот каждого номинала неограничен. Помогите Национальному банку решить эту задачу.

Формат входных данных

Первая строка входных данных содержит натуральное число N не превосходящее 100 — количество номиналов банкнот в обращении. Вторая строка входных данных содержит N различных натуральных чисел x_1, x_2, \dots, x_N , не превосходящих 10^6 — номиналы банкнот. Третья строка содержит натуральное число S , не превосходящее 10^6 — сумму, которую необходимо выдать.

Формат выходных данных

В первую строку выходного файла выведите минимальное число слагаемых (или -1, если такого представления не существует). Во вторую строку выведите это представление в любом порядке.

Пример

atm.in	atm.out
5 1 3 7 12 32 40	3 32 7 1

Задача К. Три буквы

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

Напомним, что строка $B = b_1b_2b_3 \dots b_m$, является подпоследовательностью строки $A = a_1a_2a_3 \dots a_n$, если существует строго возрастающая последовательность $\{i_1, i_2, i_3, \dots, i_m\}$ индексов A , такая, что для всех $j \in \{1, \dots, m\}$, выполняется $A_{i_j} = B_j$. Например, $B = "aba"$ является подпоследовательностью строки $A = "abacaba"$. Последовательность индексов в этом случае может быть такой: $\{1, 2, 3\}$.

Пусть Вам дана строка S состоящая только из маленьких букв латинского алфавита. Ваша задача заключается в том, чтобы посчитать количество ее подпоследовательностей «abc».

Формат входных данных

В единственной строке входного файла записана строка S , длиной не более 100 000 символов.

Формат выходных данных

В выходной файл выведите ответ на задачу.

Примеры

стандартный ввод	стандартный вывод
abc	1
ab	0