

Задача 1. Итоги олимпиады

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Школьники из кружка по информатике «Бурундучки программируют» участвуют в олимпиаде. По итогам олимпиады i -й школьник набрал a_i баллов.

Чтобы поощрить участников, руководитель кружка Михаил Юрьевич решил раздать школьникам конфеты. Для всех i и j , если i -й школьник набрал больше баллов, чем j -й, то руководитель даёт i -му школьнику $a_i - a_j$ конфет.

Помогите руководителю понять, сколько суммарно конфет ему необходимо подготовить для раздачи школьникам.

Формат входных данных

В первой строке входных данных задается число n — количество школьников ($1 \leq n \leq 500\,000$).

Вторая строка содержит n целых чисел a_i — результаты участников кружка на олимпиаде ($0 \leq a_i \leq 10^7$).

Формат выходных данных

Выведите единственное число — общее количество конфет, которое необходимо подготовить, чтобы раздать школьникам.

Обратите внимание, что ответ в этой задаче может превышать возможное значение 32-битной целочисленной переменной, поэтому необходимо использовать 64-битные целочисленные типы данных (тип `int64` в языке Pascal, тип `long long` в C++, тип `long` в Java и C#).

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи
1	15	$1 \leq n \leq 1\,000$	
2	5	Все a_i одинаковые	
3	5	Для любых $i \neq j$ выполнено $a_i \neq a_j$, также $1 \leq a_i \leq n$	
4	10	$0 \leq a_i \leq 1$	
5	15	$0 \leq a_i \leq 100$	4
6	15	Среди a_i присутствует не более двух различных значений	2, 4
7	35	—	1–6

Примеры

стандартный ввод	стандартный вывод
5 1 2 3 4 5	20
10 0 0 0 0 0 10000000 0 0 0 0	90000000

Замечание

В первом примере первый школьник не получит конфет, второй школьник получит 1 конфету, третий школьник получит $1 + 2 = 3$ конфеты, четвертый школьник получит $1 + 2 + 3 = 6$ конфет, пятый школьник получит $1 + 2 + 3 + 4 = 10$ конфет.

Задача 2. Хромой король

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Хромой король передвигается по клетчатой доске размером $n \times m$, каждый раз переходя из текущей клетки в соседнюю по стороне. Будем задавать клетку в ряду x столбце y как (x, y) .

Хромой король должен посетить все клетки, побывав в каждой клетке ровно один раз, и вернуться в начальную клетку. При этом на доске выделены две соседние клетки: (x_1, y_1) , (x_2, y_2) . В обходе доски королем клетки (x_1, y_1) и (x_2, y_2) должны встречаться подряд: оказавшись в одной из них, он должен сразу же перейти в другую.

Найдите подходящий порядок обхода доски или выясните, что его не существует.

Формат входных данных

Первая строка содержит два числа n и m ($2 \leq n, m \leq 1000$) — размеры доски.

Вторая строка содержит четыре числа x_1, y_1, x_2, y_2 — координаты двух соседних клеток ($1 \leq x_1, x_2 \leq n; 1 \leq y_1, y_2 \leq m; |x_1 - x_2| + |y_1 - y_2| = 1$).

Формат выходных данных

Если такого обхода доски не существует, требуется вывести одно число -1 .

Иначе выведите $n \times m + 1$ пару чисел — координаты клеток в порядке обхода, начальную клетку необходимо вывести дважды, в начале и в конце.

Система оценки

В этой задаче 50 тестов, каждый оценивается независимо в 2 балла.

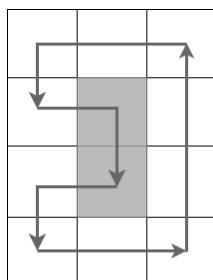
В этой задаче во время тура вам сообщается результат проверки на каждом тесте.

Примеры

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

Замечание

На рисунке показан обход доски для первого примера.



Задача 3. Расстановки фишек

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

Дана квадратная доска размера $m \times m$. Строки и столбцы доски пронумерованы от 1 до m .

Следует расставлять на доске фишки так, чтобы в каждой клетке находилось не более одной фишки. При этом должны выполняться n ограничений. В i -м ограничении заданы два целых числа r_i и c_i , означающие, что в прямоугольнике, состоящем из клеток с координатами $[1 \dots r_i] \times [1 \dots c_i]$, может находиться не более одной фишки.

Необходимо найти остаток от деления количества различных расстановок фишек, удовлетворяющих всем ограничениям, на $10^9 + 7$.

Формат входных данных

Первая строка входных данных содержит целые числа n и m — количество ограничений и размер доски ($1 \leq n \leq 2 \cdot 10^5$, $1 \leq m \leq 10^9$).

Далее следуют n строк, в каждой из которых записаны два числа r_i и c_i ($1 \leq r_i, c_i \leq m$).

Формат выходных данных

Требуется вывести одно число — количество допустимых расстановок фишек, взятое по модулю $10^9 + 7$.

Система оценки

Подзадача	Баллы	Доп. ограничения	Необх. подзадачи
1	3	$n \leq 10, m \leq 4$	—
2	6	$n = 1, m \leq 1000$	—
3	8	$n \leq 10, m \leq 1000$	1, 2
4	8	$n \leq 15, m \leq 10^9$	1–3
5	10	$n \leq 2500, m \leq 100$	1
6	10	$n \leq 2500, m \leq 250$	1, 5
7	10	$n \leq 2500, m \leq 1000$	1–3, 5, 6
8	10	$n \leq 2500, m \leq 10^5$	1–3, 5–7
9	15	$n \leq 2 \cdot 10^5, m \leq 2 \cdot 10^5$	1–3, 5–8
10	20	нет	1–9

Примеры

стандартный ввод	стандартный вывод
1 4 4 4	17
2 2 1 2 2 1	10
3 5 2 5 3 4 4 4	4480

Замечание

В первом примере на всей доске может быть поставлено не более одной фишки. Есть $4 \times 4 = 16$ вариантов поставить одну фишку и 1 вариант с нулём расставленных фишек.

Задача 4. Прыжки по вершинам

Ограничение по времени: 1 секунда
Ограничение по памяти: 512 мегабайт

В компьютерной игре «Суперскок» герой прыгает между вершинами горной цепи с целью попасть на точку с флагом, где завершается уровень.

Горная цепь на уровне состоит из n подряд идущих зубцов, i -й из которых находится в позиции i и имеет высоту h_i . При этом для любых $i < j$ герой может прыгнуть по прямой с зубца i на зубец j , при условии, что во время полёта по прямой на его пути не будет других зубцов. Более формально, не найдётся такого k , что $i < k < j$ и вершина k -го зубца — точка с координатами (k, h_k) — находится строго выше отрезка, соединяющего точки (i, h_i) и (j, h_j) .

Компания «Вместе с ИИ» занимается тренировкой нейросети для управления героем в этой игре. Для создания тренировочных данных необходимо ответить на несколько запросов: для пары индексов l, r ($1 \leq l \leq r \leq n$) выяснить, за какое минимальное число прыжков, начав на зубце с номером l , герой сможет попасть на зубец с номером r .

Формат входных данных

В первой строке входных данных находится число n ($1 \leq n \leq 10^5$) — число зубцов.

Во второй строке находятся n чисел: h_1, h_2, \dots, h_n ($0 \leq h_i \leq 10^{12}$) — высоты зубцов.

В третьей строке находится число q ($1 \leq q \leq 10^5$) — количество запросов.

В каждой из следующих q строк находятся два числа l_i, r_i ($1 \leq l_i \leq r_i \leq n$) — параметры очередного запроса.

Формат выходных данных

Для каждого запроса в отдельной строке выведите целое неотрицательное число — минимальное необходимое число прыжков.

Система оценки

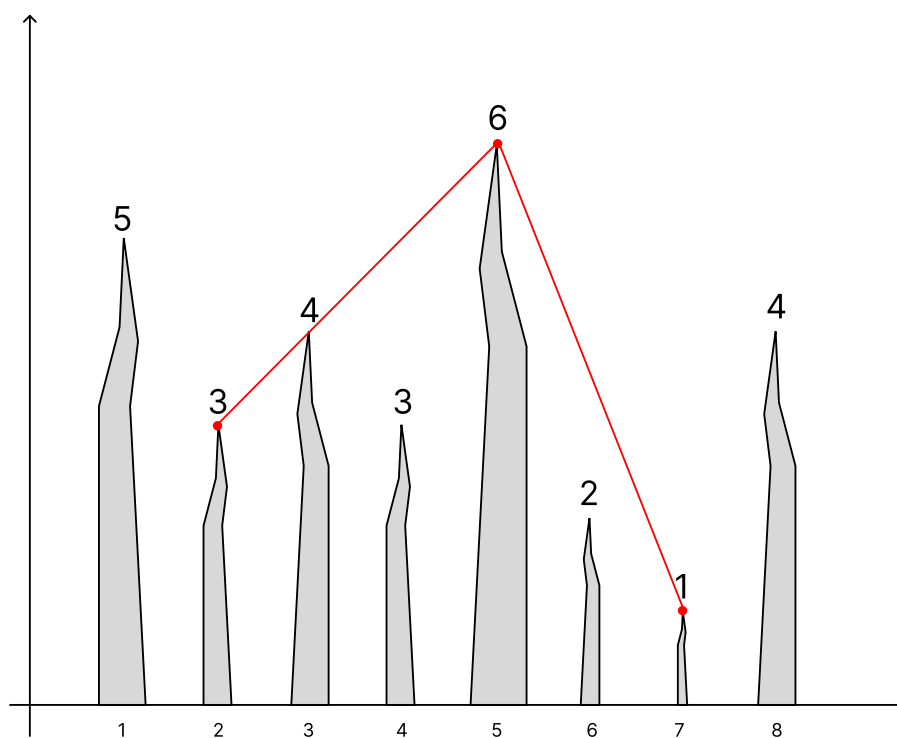
Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи
1	9	$n, q \leq 300$	
2	9	$n, q \leq 5000$	1
3	14	$h_i \leq 10$	
4	21	Существует k , такое что для всех i выполнено $l_i \leq k \leq r_i$	
5	27	$n, q \leq 5 \cdot 10^4$	1, 2
6	20	—	1–5

Пример

стандартный ввод	стандартный вывод
8	2
5 3 4 3 6 2 1 4	2
3	0
1 8	
2 7	
4 4	

Замечание

Разберём второй запрос в примере из условия. Путь героя от зубца 2 до зубца 7 может выглядеть следующим образом:



Он посетит вершины 2, 5 и 7, совершив два прыжка.