

## Задача 1. Итоги олимпиады

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

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

Чтобы поощрить участников, руководитель кружка Александр Сергеевич 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 перемещается 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.

Выведите 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, выведите 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.

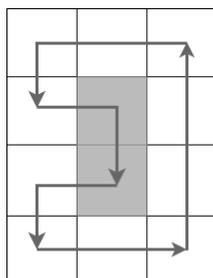
В этой задаче во время тура вам сообщается результат проверки на каждом тесте.

### 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  $m$ .

Требуется 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.

Определите 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

Выведите 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 «Мегапрыжок» герой прыгает между вершинами горной цепи с целью попасть на точку с флагом, где завершается уровень.

Горная цепь в игре 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 «Победи 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

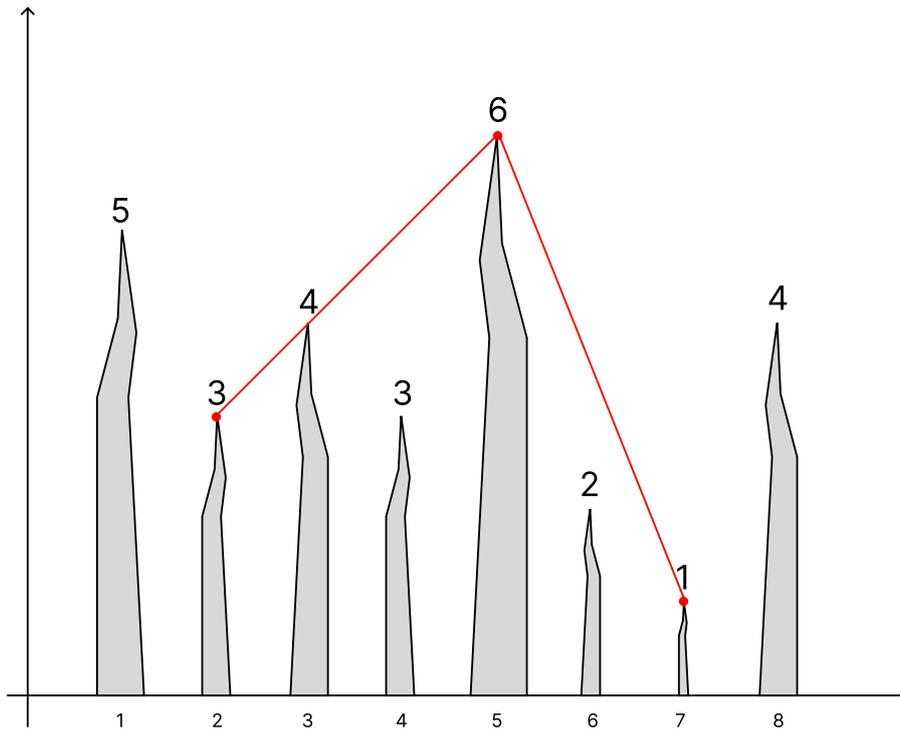
Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи
1	9	$n, q \leq 300$	
2	9	$n, q \leq 5000$	1
3	14	$h_i \leq 10$	
4	21	Существует $k$ , такое что для всех $i$ выполнено $l_i \leq k \leq r_i$	
5	27	$n, q \leq 5 \cdot 10^4$	1, 2
6	20	—	1–5

## 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.

## Задача 5. Покраска бруска

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

На фабрике изготавливают цветные кубики. Для этого берётся заготовка — деревянный брусок  $a \times b \times c$ . Сначала его распиливают на  $a \cdot b \cdot c$  единичных кубиков, а потом каждый кубик окрашивается со всех сторон.

Однако из-за ошибки в программе для станка, написанной с помощью системы вайб-кодинга «Кодер 239», в этот раз всё произошло наоборот: сначала стороны бруска были покрашены со всех сторон, а затем он был распилен на единичные кубики. Из-за этого у разных кубиков в этой партии могло оказаться разное количество покрашенных сторон.

Для оценки ущерба необходимо посчитать количество кубиков, у которых покрашено ровно  $k$  сторон.

### Формат входных данных

Единственная строка содержит четыре числа:  $a, b, c$  ( $1 \leq a, b, c \leq 10^5$ ) — размеры бруска, — и число покрашенных сторон кубика  $k$  ( $0 \leq k \leq 6$ ).

### Формат выходных данных

Выведите одно число — количество единичных кубиков с заданным числом покрашенных сторон.

### Система оценки

В этой задаче 20 тестов, каждый оценивается независимо в 5 баллов.

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

### Примеры

стандартный ввод	стандартный вывод
3 3 3 3	8
4 2 1 3	4

## Задача 6. Битовая магия

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

Даны три неотрицательных целых числа  $b$ ,  $l$  и  $r$ , записанные в шестнадцатеричной системе счисления.

Напомним, что шестнадцатеричная система счисления (основание 16) использует цифры 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, где A соответствует числу 10, B — 11, C — 12, D — 13, E — 14, F — 15. Например, число 1F в шестнадцатеричной системе равно  $1 \cdot 16 + 15 = 31$  в десятичной системе.

Операция  $\&$  обозначает побитовое AND (побитовое «И») над двоичными представлениями чисел. Рассмотрим двоичные записи чисел  $x$  и  $b$ , при необходимости дополним их слева нулями до равной длины. Для каждого разряда  $i$ :

$$(x \& b)_i = \begin{cases} 1, & \text{если } x_i = 1 \text{ и } b_i = 1, \\ 0, & \text{иначе.} \end{cases}$$

То есть в каждом бите результат равен 1 тогда и только тогда, когда в этом бите у обоих чисел стоит 1.

Определите количество целых чисел  $x$ , таких, что  $l \leq x \leq r$  и выполняется условие  $x \& b = b$ . Выведите остаток от деления этого количества на  $10^9 + 7$ .

### Формат входных данных

Во входных данных даны три строки: первая строка содержит число  $l$ , вторая строка содержит число  $r$ , третья строка содержит число  $b$ .

Каждое число задано в шестнадцатеричной системе счисления без ведущих нулей (кроме случая самого числа 0) и состоит из символов 0-9, A-F. Длина каждой строки не превосходит 50 000 символов. Гарантируется, что  $0 \leq l \leq r$ .

### Формат выходных данных

Выведите одно целое число — количество значений  $x$ , для которых выполняются условия задачи, по модулю  $10^9 + 7$ . Ответ выведите в десятичной системе счисления без ведущих нулей.

### 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	10	$0 \leq r, b < 16^4, l = 0$	
2	5	$0 \leq l, r, b < 16^4$	1
3	10	$0 \leq r, b < 16^7, l = 0$	1
4	6	$0 \leq l, r, b < 16^7$	1-3
5	10	$0 \leq r, b < 16^{15}, l = 0$	1, 3
6	7	$0 \leq l, r, b < 16^{15}$	1-5
7	14	$0 \leq r, b < 16^{1000}, l = 0$	1, 3, 5
8	7	$0 \leq l, r, b < 16^{1000}$	1-7
9	11	$0 \leq r, b < 16^{50000}, l = 0$	1, 3, 5, 7
10	12	$0 \leq l, r < 16^{50000}, b = 0$	
11	8	$0 \leq l, r, b < 16^{50000}$	1-10

## Примеры

стандартный ввод	стандартный вывод
8 F 5	2
2 F9 A	60

## Замечание

В первом примере из условия подходящими значениями  $x$  являются шестнадцатеричные числа D и F.

## Задача 7. Скользящие окна

Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Рассмотрим массив чисел  $b_1, \dots, b_m$ . Скользящими окнами длины  $k$  ( $k \leq m$ ) на этом массиве назовем все подотрезки длины  $k$ , то есть отрезки  $[b_1, \dots, b_k], [b_2, \dots, b_{k+1}], \dots, [b_{m-k+1}, \dots, b_m]$ .

Дан массив чисел  $a_1, \dots, a_n$  длины  $n$ .

Необходимо ответить на  $q$  запросов следующего вида про этот массив: для заданных  $l, r, k$  найти сумму минимумов на скользящих окнах длины  $k$  на подотрезке  $[a_l, \dots, a_r]$ .

### Формат входных данных

В первой строке входных данных даны два целых числа  $n$  и  $q$  ( $1 \leq n, q \leq 100\,000$ ) — длина массива и количество запросов.

Во второй строке даны  $n$  целых чисел  $a_1, \dots, a_n$  ( $1 \leq a_i \leq 10^9$ ) — значения чисел в массиве.

В следующих  $q$  строках даны запросы. В  $i$ -й из них даны три целых числа  $l_i, r_i$  и  $k_i$  ( $1 \leq l \leq r \leq n$ ,  $1 \leq k \leq r - l + 1$ ) — левая и правая границы отрезков и длина скользящего окна в  $i$ -м запросе.

### Формат выходных данных

Выведите  $q$  строк с ответами на запросы. В  $i$ -й строке выведите единственное число — сумму минимумов на скользящих окнах длины  $k_i$  на подотрезке  $[a_{l_i}, \dots, a_{r_i}]$ .

### Система оценки

Подзадача	Баллы	Дополнительные ограничения	Необх. подзадачи
1	6	$n, q \leq 300$	
2	12	$n, q \leq 4000$	1
3	8	$n, q \leq 10\,000$	1, 2
4	11	$n \leq 4\,000$	1, 2
5	10	$k_i$ равны во всех запросах	
6	14	$a_i \leq 2$	
7	7	$a_i \leq 20$	6
8	15	$l_i = 1, r_i = n$	
9	17	нет	1–8

### Пример

стандартный ввод	стандартный вывод
6 3	4
4 6 1 2 5 3	9
2 5 2	1
2 4 1	
1 6 6	

## Задача 8. XOR Раскраска

Ограничение по времени: 2 секунды  
Ограничение по памяти: 512 мегабайт

Даны два массива неотрицательных целых чисел  $A = [a_1, a_2, \dots, a_n]$  и  $B = [b_1, b_2, \dots, b_m]$ .

Пусть  $S(i) = \{j | (a_i \oplus b_j) \leq x\}$ . Иными словами,  $S(i)$  это множество всех индексов  $j$  массива  $B$ , для которых побитовое исключающее или  $a_i$  и  $b_j$  не превосходит  $x$ .

Требуется найти минимальное число  $k$ , чтобы можно было покрасить элементы массива  $A$  в  $k$  цветов таким образом, что если  $S(x)$  и  $S(y)$  пересекаются, то  $x$  и  $y$  покрашены в разный цвет.

Иначе говоря, можно найти такие  $c_1, c_2, \dots, c_n$ , что  $1 \leq c_i \leq k$ , и при этом если  $S(x) \cap S(y) \neq \emptyset$ , то  $c_x \neq c_y$ .

Напомним, что побитовое «исключающее или» ( $\oplus$ , xor) двух целых неотрицательных чисел определяется следующим образом: запишем оба числа в двоичной системе счисления,  $i$ -й двоичный разряд результата равен 1, если ровно у одного из аргументов он равен 1. Например,  $(14 \text{ xor } 7) = (1110_2 \oplus 0111_2) = 1001_2 = 9$ . Эта операция реализована во всех современных языках программирования, в языках C++, Java и Python она записывается как «^», в Паскале как «xor».

### Формат входных данных

Входные данные для этой задачи содержат несколько тестовых примеров.

Первая строка ввода содержит одно целое число  $t$  ( $1 \leq t \leq 100$ ) — количество тестовых примеров.

Далее следуют описания тестовых примеров.

В первой строке каждого тестового примера записаны три целых числа  $n$ ,  $m$  и  $x$  ( $1 \leq n, m \leq 500\,000$ ,  $0 \leq x < 2^{30}$ ).

Во второй строке записаны  $n$  целых чисел  $a_1, a_2, \dots, a_n$  — элементы массива  $A$  ( $0 \leq a_i < 2^{30}$ ).

В третьей строке записаны  $m$  целых чисел  $b_1, b_2, \dots, b_m$  — элементы массива  $B$  ( $0 \leq b_i < 2^{30}$ ).

Гарантируется, что как сумма значений  $n$ , так и сумма значений  $m$  по всем тестовым примерам не превосходит 500 000.

### Формат выходных данных

Для каждого тестового примера выведите одно целое число — минимальное искомое  $k$ .

### Система оценки

Подзадача	Баллы	Доп. ограничения	Необх. подзадачи
1	5	$n \leq 2$	—
2	5	$n \leq 5$	1
3	5	$n \leq 15$	1,2
4	5	$n \leq 100$	1–3
5	5	$n \leq 2\,000$	1–4
6	10	$n \leq 5\,000$	1–5
7	5	$n \leq 100\,000$ , $m = 2$	—
8	10	$n \leq 100\,000$ , $m = 3$	—
9	5	$n, m \leq 100\,000$ ; $a_i, b_i, k < 2$	—
10	10	$n, m \leq 100\,000$ ; $a_i, b_i, k < 4$	9
11	35	нет	1–10

## Пример

стандартный ввод	стандартный вывод
3	1
2 2 0	4
0 0	5
1 1	
5 5 3	
0 1 2 3 4	
0 1 2 3 4	
5 5 4	
0 1 2 3 4	
0 1 2 3 4	