

## Задача 5. Покраска бруска

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

На фабрике создают 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 «Умный кодер», 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

Вывод должен содержать 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 (с основанием системы 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.

Требуется определить 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

Вывод должен содержать 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	$0 \leq r, b < 16^{50\,000}, l = 0$	1, 3, 5, 7
10	12	$0 \leq l, r < 16^{50\,000}, b = 0$	
11	8	$0 \leq l, r, b < 16^{50\,000}$	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 являются 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  $n$ .

Требуется дать ответ 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

Вывод должен содержать  $q$  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 6 1 2 5 3 2 5 2 2 4 1 1 6 6	4 9 1

## 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 определить 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 можно найти 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 вывод должен содержать 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 2 2 0 0 0 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	1 4 5