

## Задача А. Подарок

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	1 секунда
Ограничение по памяти:	256 мегабайт

Женя подарил Антону лист бумаги, на котором было написано в ряд  $n$  целых чисел.

Антон, как наиболее перспективный программист XXI века, решил систематизировать свои знания о подарке. Для этого он нарисовал квадратную таблицу, в которой было  $n$  строк и  $n$  столбцов. В ячейку на пересечении  $i$ -го столбца и  $j$ -й строки он записал  $\min(a_i, a_j)$ , где  $a_i$  —  $i$ -е число, записанное на листе, а  $a_j$  —  $j$ -е число.

После этого Антон покрасил таблицу в шахматном порядке. Он красил клетку на пересечении  $i$ -го столбца и  $j$ -й строки в черный цвет, если  $(i + j)$  оказывалось нечётно.

После всех этих операций Антоша очень устал, поэтому он попросил вас посчитать **абсолютную** разность суммы чисел в черных клетках и суммы чисел в белых клетках.

### Формат входных данных

В первой строке находится число  $n$  — длина массива.  $1 \leq n \leq 10^5$ .

Во второй строке находятся  $n$  чисел — сам массив  $a$ .  $1 \leq a_i \leq 10^6$ .

### Формат выходных данных

Выведите одно число — разность суммы чисел, записанных в черных клетках, и суммы чисел, записанных в белых клетках.

### Система оценки

В данной задаче 25 тестов (помимо тестов из условия), каждый из них оценивается из 4 баллов. Результаты работы ваших решений на всех тестах будут доступны сразу во время соревнования.

- Решения, работающие при  $n \leq 10$  будут набирать не менее 10 баллов.
- Решения, работающие при  $a_i = 1$  будут набирать не менее 10 баллов.
- Решения, работающие при  $a_i \leq 2$  будут набирать не менее 40 баллов
- Решения, работающие при  $n \leq 1000$  будут набирать не менее 20 баллов.

### Примеры

стандартный ввод	стандартный вывод
3 1 2 3	2
10 1 2 3 4 5 6 7 8 9 10	5

### Замечание

В первом примере Антон получит такую табличку:

1	1	1
1	2	2
1	2	3

Разность равна  $(1 + 1 + 1 + 2 + 3) - (1 + 1 + 2 + 2) = 2$ .

**Обратите внимание**, что входные данные, а также ответ могут быть достаточно большими, поэтому следует использовать 64-битный тип данных, например `long long` в C/C++, `long` в Java, `int64` в Pascal.

## Задача В. Гонка

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

Совсем недавно компания «Mentendo» выпустила новую версию культовой игры «Super Mario Kart»! В новой версии игры, как и в оригинальной, герои участвуют в гонках на машинах.

Всего в игре есть  $n$  персонажей, пронумерованных целыми числами от 1 до  $n$ . Перед выпуском игры разработчики провели опрос фанатов, и выяснили, насколько они любят каждого из персонажей. По итогам опроса для каждого персонажа была выявлена его *популярность* — целое неотрицательное число. Популярность  $i$ -го персонажа равна  $b_i$ .

Каждая игра состоит из нескольких заездов, в каждом из которых принимают участие все персонажи. По итогам каждого заезда лучшие  $k$  персонажей получают некоторое количество баллов в общий зачет. А именно, персонаж, финишировавший первым, получает  $a_1$  баллов, персонаж, финишировавший вторым, получает  $a_2$  баллов, и так далее. Персонаж, финишировавший  $k$ -м, получает  $a_k$  баллов в общий зачет. Все остальные персонажи получают 0 баллов в общий зачет по итогам заезда. Благодаря современным технологиям, время финиша измеряется абсолютно точно, а потому можно считать, что никакие два персонажа не финишируют одновременно. Также, исходя из принципа справедливого распределения баллов, выполняется неравенство  $a_1 > a_2 > \dots > a_k > 0$ .

Для подведения итогов игры для каждого из персонажей вычисляется его *общий балл* в этой игре. Общий балл персонажа определяется следующим образом: берутся все баллы (в том числе и нулевые), набранные этим персонажем за все заезды, из этих баллов вычеркивается  $s$  минимальных значений, а оставшиеся значения складываются и прибавляются к популярности персонажа. Например, если Луиджи обладает популярностью 3, в четырех заездах он набрал 2, 1, 3 и 0 баллов соответственно, а  $s = 2$ , то из баллов, полученных за заезды, будут вычеркнуты два минимальных, то есть 1 и 0, а общий балл Луиджи будет равен  $3 + 2 + 3 = 8$ . Итоги игр, в которых было менее  $s$  заездов, не подводятся, чтобы избежать неоднозначности трактовки правил.

Во время очередной игры Марио и динозаврик Йоши решили немного пофантазировать. За время игры уже было проведено  $m$  заездов, результаты которых Марио и Йоши знают. Пока готовится очередной заезд, герои играют в следующую мини-игру: Йоши называет номера двух различных персонажей  $u$  и  $v$ , а Марио должен ответить, какое минимальное число заездов нужно ещё провести, чтобы общий балл персонажа  $v$  был строго больше общего балла персонажа  $u$ . Обратите внимание, что герои лишь фантазируют, а Марио интересуется теоретический минимум количества дополнительных заездов, то есть Марио может выбрать самый выгодный для себя исход каждого из дополнительных заездов. Мини-игра показалась Марио довольно скучной, поэтому он просит вас написать программу, которая могла бы играть с Йоши вместо него.

### Формат входных данных

Первая строка входных данных содержит четыре целых числа  $n$ ,  $m$ ,  $k$ ,  $s$  — количество персонажей в игре, количество уже проведенных заездов, количество персонажей, которые получают ненулевые баллы в общий зачет по итогам одного заезда, и количество заездов, не учитываемых при подведении итогов игры, соответственно ( $2 \leq n \leq 1000$ ,  $1 \leq m \leq 1000$ ,  $1 \leq k \leq n$ ,  $0 \leq s \leq \min(10, m)$ ).

Вторая строка содержит  $k$  чисел  $a_1, a_2, \dots, a_k$  — баллы, которые получают лучшие  $k$  персонажей по итогам каждого заезда ( $1 \leq a_k < a_{k-1} < \dots < a_2 < a_1 \leq 10^9$ ).

Третья строка содержит  $n$  чисел  $b_1, b_2, \dots, b_n$  — значения популярности каждого из персонажей ( $0 \leq b_1, b_2, \dots, b_n \leq 10^9$ ).

Следующие  $m$  строк описывают результаты уже состоявшихся заездов. Каждая из них содержит  $n$  различных чисел от 1 до  $n$  — список номеров персонажей в том порядке, в котором они финишировали в очередном заезде.

Следующая строка содержит единственное целое число  $q$  — количество вопросов, заданных Йоши ( $1 \leq q \leq 10^5$ ).

Каждая из следующих  $q$  строк содержит два целых числа  $u$  и  $v$  — номера персонажей, фигурирующих в очередном вопросе ( $1 \leq u, v \leq n$ ,  $u \neq v$ ).

## Формат выходных данных

Для каждого вопроса выведите единственное целое число — минимальное количество дополнительных заездов, которое необходимо провести, чтобы была теоретическая возможность того, что у персонажа с номером  $v$  общий балл по итогам всех заездов будет больше, чем у персонажа с номером  $u$ . Если дополнительных заездов проводить не надо вообще, в качестве ответа на вопрос выведите число 0.

Гарантируется, что ответ на каждый вопрос существует, то есть всегда имеется теоретическая возможность того, что персонаж с номером  $v$  через конечное число дополнительных заездов будет иметь больший общий балл, чем персонаж с номером  $u$ .

## Система оценки

Эта задача состоит из шести подзадач. Для подзадач выполняются дополнительные ограничения, указанные в таблице ниже. Для получения баллов за подзадачу необходимо пройти все тесты данной подзадачи, а также все тесты всех необходимых подзадач. Необходимые подзадачи также указаны в таблице.

Подзадача	Баллы	Ограничения					Необходимые подзадачи
		$n, m$	$q$	$s$	$b_i$	$a_i$	
1	12	$n, m \leq 50$	$q \leq 50$	$s = 0$	$b_i = 0$	$a_i \leq 10^5$	–
2	14	$n, m \leq 50$	$q \leq 50$	$s \leq \min(10, m)$	$b_i = 0$	$a_i \leq 10^5$	1
3	15	$n, m \leq 1000$	$q \leq 1000$	$s = 0$	$b_i = 0$	$a_i \leq 10^9$	1
4	17	$n, m \leq 1000$	$q \leq 1000$	$s \leq \min(10, m)$	$b_i = 0$	$a_i \leq 10^9$	1, 2, 3
5	20	$n, m \leq 1000$	$q \leq 10^5$	$s = 0$	$b_i \leq 10^9$	$a_i \leq 10^9$	1, 3
6	22	$n, m \leq 1000$	$q \leq 10^5$	$s \leq \min(10, m)$	$b_i \leq 10^9$	$a_i \leq 10^9$	1, 2, 3, 4, 5

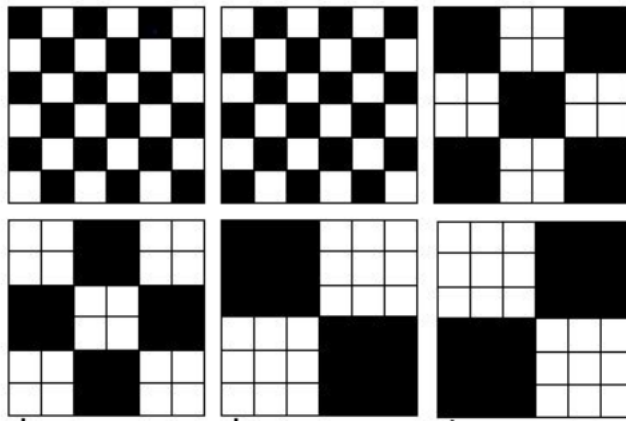
## Пример

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

## Задача С. Шахматы

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

У Жени есть клеточная доска  $N \times N$ ,  $K$  подпрямоугольников которой покрашены в черный цвет, а остальные клетки белые. Ряды доски пронумерованы от 1 до  $N$  сверху-вниз, а столбцы от 1 до  $N$  слева-направо. Доска называется **шахматной**, если ее можно разделить на несколько одинаковых квадратов со стороной  $L$ ,  $1 \leq L < N$ , что внутри каждого квадрата все клетки покрашены в один цвет, а любые два соседние по стороне квадрата имеют различные цвета. Ниже представлены все шахматные доски для  $N = 6$ .



За одно перекрашивание вы можете поменять цвет любого квадратика доски размерами  $1 \times 1$  на противоположный. Найдите наименьшее число перекрашиваний, достаточное для того, чтобы сделать описанную во вводе доску шахматной.

### Формат входных данных

В первой строке ввода находятся числа  $N, K$  ( $2 \leq N \leq 10^5, 1 \leq K \leq 10^5$ ) — размеры доски и количество покрашенных подпрямоугольников. В каждой из следующих  $K$  строк находятся 4 числа  $x_1, y_1, x_2, y_2$  ( $1 \leq x_1 \leq x_2 \leq N, 1 \leq y_1 \leq y_2 \leq N$ ) — координаты левого верхнего и правого нижнего углов очередного прямоугольника. **Гарантируется, что никакая пара подпрямоугольников не пересекается.**

### Формат выходных данных

Выведите одно число — ответ на задачу.

### Система оценки

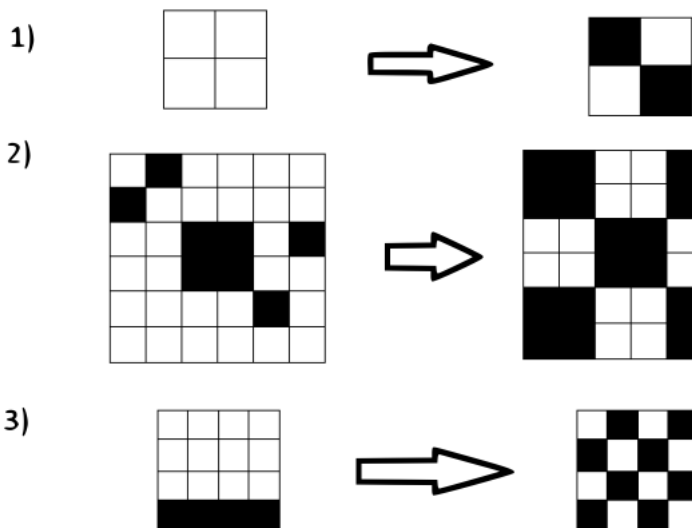
Обозначим за  $S$  максимальное количество клеток в подпрямоугольнике из ввода.

группа	ограничения	баллы
1	$2 \leq N \leq 100, K = 0$	8
2	$N$ является простым числом, $S = 1$	8
3	$2 \leq N \leq 100, 0 \leq K \leq 1000, S = 1$	15
4	$2 \leq N \leq 1000, S = 1$	16
5	$2 \leq N \leq 10^5, S = 1$	23
6		30

## Примеры

стандартный ввод	стандартный вывод
2 0	2
6 8 3 3 3 3 1 2 1 2 3 4 3 4 5 5 5 5 4 3 4 3 4 4 4 4 2 1 2 1 3 6 3 6	14
4 1 4 1 4 4	8

## Замечание



## Задача D. «Больше»-«Меньше» 2.0

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	3 секунды
Ограничение по памяти:	256 мегабайт

Ваня загадал число  $x$ , где  $x$  — целое число 0 до  $n$  включительно. Вы пытаетесь угадать Ванино число. Для этого вы фиксируете какую-то перестановку  $p$  длины  $n$  (выписываете числа от 1 до  $n$  в каком-то порядке) и по очереди задаете вопросы про них Ване. Когда вы спрашиваете про число  $q$ , Ваня отвечает вам «Больше», если  $x < q$ , и «Меньше» иначе. Однако вы не задаете бесполезные вопросы, то есть вы не спрашиваете про  $i$ , если вы уже спрашивали про  $j < i$  и получили ответ «Больше», или если вы спрашивали про  $j > i$  и получила ответ «Меньше». Несложно показать, что вы сможете однозначно восстановить  $x$  по ответам Вани.

Для фиксированного  $x$  определим  $f(p, x)$  как количество индексов  $i$ , что ответ на вопрос  $i$  был «Больше», а ответ на вопрос  $i + 1$  был «Меньше» (рассматриваются только вопросы, на которые вы получили ответы), если Вы используете перестановку  $p$ .

В этой задаче вы знаете значения чисел  $n$  и  $x$ , но не знаете перестановку. Вам необходимо найти сумму  $f(p, x)$  по всем перестановкам  $p$  длины  $n$  по модулю  $10^9 + 7$ .

### Формат входных данных

В единственной строке ввода находятся числа  $n, x$  ( $1 \leq n \leq 5000, 0 \leq x \leq n$ ).

### Формат выходных данных

Выведите единственное число — ответ на задачу.

### Система оценки

группа	ограничения	баллы
1	$n \leq 10$	10
2	$n \leq 20$	10
3	$n \leq 50$	10
4	$n \leq 500$	30
5	$n \leq 2000$	20
6	$n \leq 5000$	20

### Примеры

стандартный ввод	стандартный вывод
4 2	17
60 10	508859913