Не доверяйте свиткам

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

Это задача с двойным запуском. Ваше решение будет запущено дважды.

За игровым столом четыре игрока — мастер игры, вы (паладин) и два ваших друга — волшебник и плут. Ваш отряд в данный момент проводит инвентаризацию; оказалось, что десятки ненужных вам подобранных по пути предметов снаряжения, которыми забиты ваши рюкзаки, мешает эффективно перемещаться по лесу.

В своем рюкзаке среди добытого лута волшебник нашел свиток с отчетностью о тратах отряда, которую он вел последние n дней (вы подозревали, что плут втайне кладет пару лишних монет в свой карман). Известно, что

- ваш отряд расходует золото очень ответственно, поэтому в каждый из n дней было потрачено одинаковое число монет;
- все траты разделены на три категории: еда, снаряжение и вечеринки в таверне.

Иными словами, на свитке были записаны три массива неотрицательных целых чисел a (траты на еду), b (траты на снаряжение) и c (траты на вечеринки), каждый длины n, и известно, что $a_i + b_i + c_i = a_j + b_j + c_j$ для всех i и j.

Внимательно посмотрев на свиток, волшебник заметил, что с ним что-то не так: массивы были разных размеров, и условие на сохранение суммы соответствующих элементов трех массивов перестало выполняться. Произнеся заклинание «Обнаружение магии» и бросив кубик, волшебник установил, что это «Проклятый свиток сжатия», который удаляет из массивов подряд идущие одинаковые элементы. Например, если на свитке были записаны $a = [2, 2, 2, 1, 5]$, $b = [3, 3, 1, 2, 3]$ и $c = [4, 4, 6, 6, 1]$, то спустя некоторое время они превратятся в $a' = [2, 1, 5]$, $b' = [3, 1, 2, 3]$ и $c' = [4, 6, 1]$.

Ваш персонаж волшебнику тоже не до конца доверяет. Поэтому у вас в кармане случайно нашлась записка с сокращенной информацией о тратах S, состоящей из не более k целых чисел от 0 до $2^{30} - 1$. На удивление, этой информации хватило, чтобы восстановить как могли выглядеть исходные a, b и c.

Какую информацию S следует запомнить, чтобы по n, a', b' и c' можно было восстановить какие-то a, b и c, удовлетворяющие условию?

Формат входных данных

В первой строке ввода дано единственное целое число 1 или 2 — номер запуска.

Первый запуск

Во второй строке ввода даны два целых числа n и k — число дней в отчетности и ограничение на длину S ($1 \leqslant n \leqslant 30\,000$; $1 \leqslant k \leqslant 60\,000$).

В третьей строке перечислены n целых чисел a_i — траты отряда на еду в каждый из дней ($0 \leqslant a_i < 2^{20}$). В четвертой и пятой строках в том же формате перечислены b_i и c_i — траты на снаряжение и на вечеринки ($0 \leqslant b_i, c_i < 2^{20}$). Гарантируется, что $a_i + b_i + c_i$ — постоянная величина.

Второй запуск

Во второй строке ввода даны пять целых чисел n, m_s, $m_{a'}$, $m_{b'}$ и $m_{c'}$ — изначальное n из первого запуска, длина выведенной вашим решением при первом запуске последовательности S, а также длины a', b' и c' ($1 \leqslant m_{a'}, m_{b'}, m_{c'} \leqslant n \leqslant 30\,000$; $1 \leqslant m_s \leqslant k$).

В следующих четырех строках перечислены элементы S, a', b' и c' соответственно. Гарантируется, что S получено при первом запуске вашего решения, а a', b' и c' получены из a, b и c для первого запуска удалением повторений подряд идущих элементов.

Формат выходных данных

Первый запуск

В первой строке выведите целое число m_s от 1 до k — длину последовательности S, которую вы запишете и уберете в карман.

Во второй строке через пробел выведите m_s целых чисел от 0 до $2^{30} - 1$ — элементы S.

Второй запуск

Выведите три строки, в каждой по n чисел от 0 до $2^{20} - 1$ — элементы массивов a, b и c соответственно.

Если существует несколько возможных подходящих ответов, выведите любой из них — не обязательно восстанавливать в точности те a, b и c, которые были даны во вводе при первом запуске.

Протокол взаимодействия

Во избежание получения некорректных вердиктов вроде `Idleness Limit Exceeded` или `Security Violation` заканчивайте вывод каждой строки символом перевода строки (‘`\n`’) и сбросом буфера вывода (`cout.flush()`, `sys.stdout.flush()`, `System.out.flush()`).

Система оценки

Баллы за каждую подзадачу начисляются только в случае, если все тесты для этой подзадачи и необходимых подзадач успешно пройдены. Обратите внимание, что прохождение тестов из условия не является необходимым для тестирования на некоторых группах тестов.

В последней подзадаче 10 тестов, каждый из которых независимо оценивается от 0 до 2 баллов. Пусть решение жюри на тесте выводит S длины j. Тогда ваше решение получит

- 2 балла, если выведет S не длиннее, чем j;
- 1 балл, если выведет S длины от $j + 1$ до $j + 30$;
- 0 баллов, если выведет S длины хотя бы $j + 31$.

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
0	–	примеры из условия		полная
1	4	$n \leqslant 3$, $k = 1$		первая ошибка
2	5	$n \leqslant 10$, $k = 1$	1	первая ошибка
3	8	$n \leqslant 100$, $k = 4$	1, 2	первая ошибка
4	9	$n \leqslant 1000$, $k = 1000$	0	первая ошибка
5	4	$k = 60\,000$	0	первая ошибка
6	5	$k = 40\,000$	0, 5	первая ошибка
7	12	$k = 30\,000$	0, 5, 6	первая ошибка
8	13	$k = 15\,000$	0, 5 – 7	первая ошибка
9	7	$a_i = 0$ для всех i, $k = 1$		первая ошибка
10	13	$k = 3\,000$	0, 5 – 8	первая ошибка
11	10×2	$k = 1\,900$	0 – 8, 10	полная

Примеры

стандартный ввод	стандартный вывод
1 5 15 2 2 2 1 5 3 3 1 2 3 4 4 6 6 1 2 5 3 3 4 3 3 2 4 2 1 5 3 1 2 3 4 6 1	 3 3 2 4 2 2 1 5 5 3 1 2 3 3 4 6 6 1 1
1 8 24 4 5 7 5 7 9 7 7 3 3 2 4 4 3 5 6 7 6 5 5 3 2 2 1 2 8 1 7 6 6 11347 4 5 7 5 7 9 7 3 2 4 3 5 6 7 6 5 3 2 1	 1 11347 4 5 7 5 7 9 7 7 3 3 2 4 4 3 5 6 7 6 5 5 3 2 2 1