

## Задача А. Лабораторные работы

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

На кафедре КТ Университета ИТМО преподают курс «Алгоритмы и структуры данных». Основные активности в этом курсе — лекции, практические занятия и лабораторные работы. Лабораторные работы заключаются в решении задач из тематических констестов. Констесты устроены прямо как на этой олимпиаде, но более тематические и менее олимпиадные.

График курса очень плотный, поэтому за семестр проходит  $n$  лабораторных работ. Лабораторная с номером  $i$  длится ровно  $a_i$  дней, первая работа начинается вместе с началом курса, а каждая следующая работа начинается сразу после завершения предыдущей. Вместе с концом последней лабораторной завершается семестр.

Первокурсник Влад твердо решил совмещать учебу и все  $k - 1$  своих хобби, поэтому будет решать лабораторные каждый  $k$ -й день семестра, то есть в дни номер  $k, 2k, 3k$ , и так далее. Влад достаточно умный, поэтому если он сядет решать какую-то лабораторную, то решит в ней хотя бы одну задачу.

Найдите количество лабораторных, в которых Влад получит ненулевой результат.

### Формат входных данных

В первой строке даны два целых числа  $n$  и  $k$  — количество лабораторных работ в курсе, и с какой частотой Влад будет решать лабораторные ( $1 \leq n \leq 2 \cdot 10^5$ ;  $1 \leq k \leq 10^{18}$ ).

Во второй строке через пробел перечислены  $n$  целых чисел  $a_1, a_2, \dots, a_n$  — длительности лабораторных работ ( $1 \leq a_i \leq 10^{18}$ ).

### Формат выходных данных

Выведите единственное число — количество лабораторных работ, в которых Влад решит хотя бы одну задачу.

### Система оценки

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

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
1	5	$n, k, a_i \leq 10$		полная
2	13	$k \leq 2$		первая ошибка
3	19	сумма всех $a_i$ не превосходит $10^6$ ; $k \leq 10^6$	1	первая ошибка
4	29	$a_i \leq 3 \cdot k$ ; $k \leq 10^9$	1	первая ошибка
5	34	нет	1 – 4	первая ошибка

### Примеры

стандартный ввод	стандартный вывод
5 2 1 2 1 2 1	3
3 10 1 2 3	0
6 5 6 10 3 1 2 7	4

## Задача В. Симулятор студента

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

В коворкинге Университета ИТМО решили установить приставку, чтобы студенты могли интересно проводить перерывы между парами. Правда пока что на этой приставке доступна только одна игра — «Симулятор студента».

Как и все симуляторы, эта игра бесконечно реалистичная, поэтому в ней можно делать все, что делают студенты — в том числе, опаздывать на пары. Поле в событии «Успеть на пару» представляет собой бесконечную клетчатую решетку. Главный герой игры Гриша торопится из общежития в клетке с координатами  $(0, 0)$  в ИТМО с координатами  $(x, y)$ .

Для управления персонажем нужно использовать кнопки на джойстике. Можно переместиться на одну клетку вправо (+1 по координате  $X$ ), нажав «R», влево (−1 по координате  $X$ ), нажав «L», вверх (+1 по координате  $Y$ ), нажав «U» и вниз (−1 по координате  $Y$ ), нажав «D».

Помимо этого, игрок может нажать специальный код  $S$ . Если игрок нажал все клавиши кода подряд, а после этого нажал на кнопку «Q», то происходит  $d$  перемещений в какую-то из соседних клеток. Клетки называются соседними, если у них есть общая сторона. Например, если код равен «LRUU», то после нажатия комбинации «LRUUQ», Гриша переместится влево, вправо, и дважды вверх, после чего сделает еще  $d$  перемещений.

Ваша задача — привести Гришу в ИТМО за минимальное число нажатий кнопок. Гриша по природе очень удачливый, а еще он совсем критично опаздывает, поэтому схватится за любой шанс успеть. Иными словами, можно считать, что  $d$  перемещений после нажатия кода всегда двигают Гришу так, чтобы минимизировать количество оставшихся после этого нажатий на кнопки. Определите, какое число нажатий на кнопки понадобится.

**Заметьте**, что достаточно провести Гришу через клетку  $(x, y)$  — как только он окажется в ней, он зайдет в ИТМО и больше не будет перемещаться. Например, если код «UQ» вызывает  $d = 10$  перемещений, а требуется попасть в клетку  $(3, 0)$ , то ответ на задачу равен 2 — достаточно прожать код один раз, чтобы Гришин путь прошел через ИТМО.

### Формат входных данных

В первой строке через пробел даны три числа  $x, y$  и  $d$  — координаты конечной клетки и количество перемещений, совершаемых при нажатии кода ( $1 \leq x, y, d \leq 10^9$ ).

Во второй строке вводится сам код  $S$  ( $1 \leq |S| \leq 2 \cdot 10^5$ ). Код состоит из символов ‘R’, ‘L’, ‘U’ и ‘D’.

### Формат выходных данных

Выведите единственное число — длину кратчайшей последовательности нажатий на кнопки, приводящей Гришу в ИТМО.

### Система оценки

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

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
1	12	$x, y, d,  S  \leq 10$		полная
2	19	$x, y, d \leq 100;  S  \leq 10$	1	полная
3	22	$x, y \leq 10^6$	1, 2	первая ошибка
4	17	$ S  = 1$		первая ошибка
5	30	нет	1 — 4	первая ошибка

## Примеры

стандартный ввод	стандартный вывод
3 5 6 RDR	5
8 6 9 RLRLUDLLRU	14
4 1 10 R	2
3 5 3 LURL	8
4 5 3 UDU	9
5 5 9 RLDRU	6
3 5 3 RR	6

## Задача С. Кубки

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

В 144-й аудитории университета ИТМО на двух шкафах в ряд расставлены кубки за победы студентов кафедры КТ на олимпиадах по спортивному программированию.

На первом шкафу в ряд стоят  $n$  кубков,  $i$ -й из которых выигран на олимпиаде, которой мы условно присвоим идентификационный номер  $a_i$ . Аналогично,  $i$ -й из  $m$  кубков на втором шкафу взят на олимпиаде номер  $b_i$ . Известно, что на каждом шкафу стоят кубки за разные олимпиады, то есть  $a_i \neq a_j$  и  $b_i \neq b_j$  при  $i \neq j$ . При этом не обязательно, что все  $n + m$  кубков различны, бывает такое, что  $a_i = b_j$ . Кубки, взятые за одну и ту же олимпиаду, следует считать одинаковыми.

При реорганизации аудитории было решено оставить только один шкаф. Однако всем уже приглянулись текущие последовательности кубков на шкафах, и никто не хочет терять возможность видеть те же последовательности кубков после реорганизации.

Чтобы сохранить эстетику аудитории, было решено выбрать такую последовательность кубков  $c$ , что и  $a$ , и  $b$  входят в нее в качестве подпоследовательностей. *Подпоследовательность* — элементы на не обязательно идущих подряд индексах  $i_1 < i_2 < \dots < i_k$ . Например, последовательность  $[1, 2, 3]$  является подпоследовательностью  $[2, 1, 5, 2, 4, 6, 3]$ .

Поскольку места на одном шкафу не так много, последовательность  $c$  должна иметь как можно меньшую длину. А для большей гармоничности среди всех подходящих  $c$  минимальной длины следует выбрать ту, которая начинается на как можно меньший первый элемент.

### Формат входных данных

В первой строке входных данных даны два числа  $n$  и  $m$  — длины последовательностей  $a$  и  $b$ , соответственно ( $1 \leq n, m \leq 2 \cdot 10^5$ ).

Во второй строке через пробел перечислены  $n$  целых чисел  $a_i$  — номера олимпиад, на которых были взяты кубки на первом шкафу ( $1 \leq a_i \leq 10^9$ ). В третьей строке в том же формате перечислены  $m$  целых чисел  $b_i$  — номера олимпиад кубков на втором шкафу ( $1 \leq b_i \leq 10^9$ ).

Гарантируется, что все  $a_i$  различны и все  $b_i$  различны.

### Формат выходных данных

В первой строке выведите целое число  $k$  — минимальную длину искомой последовательности.

В следующей строке выведите элементы самой последовательности  $c$  длины  $k$  через пробел. Если существует несколько подходящих последовательностей длины  $k$ , выведите ту из них, которая начинается на меньший элемент. Среди всех таких можно выбрать любую.

### Система оценки

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

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
1	10	$n, m \leq 10, 1 \leq a[i], b[i] \leq 10$		полная
2	15	$n, m \leq 100$	1	первая ошибка
3	18	$a_i < a_{i+1}$ и $b_i < b_{i+1}$ для всех $i$		первая ошибка
4	11	$n \leq m; a_i \leq m, b_i < b_{i+1}$ для всех $i$		первая ошибка
5	19	$n, m \leq 2000$	1, 2	первая ошибка
6	27	нет	1 — 5	первая ошибка

## Примеры

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

## Задача D. Доступ к серверу

Имя входного файла:	стандартный ввод
Имя выходного файла:	стандартный вывод
Ограничение по времени:	4 секунды
Ограничение по памяти:	256 мегабайт

Чтобы загружать задачи на олимпиады в PCMS, требуется доступ к серверу, на котором сам PCMS запущен. Обычно с этим проблем не возникает, но в этот раз что-то пошло не так, и код доступа был утерян, поэтому задачи были загружены прямо перед началом конкурса.

Код доступа — массив  $a$  длины  $m$  из целых чисел от 0 до 139. Известно, что длина кода равна  $m$ . Он был зашифрован с помощью шифра Виженера ключом  $k$  длины  $q$  ( $q$  — делитель  $m$ ). Для шифрования код разбивается на блоки длины  $q$ , после чего в каждом блоке  $i$ -й символ увеличивается на  $k_i$  по модулю 139. Иными словами,  $b_{i \cdot q + j}$  при  $0 \leq i < \frac{m}{q}$  и  $1 \leq j \leq q$  заменяется на  $(b_{i \cdot q + j} + k_j) \bmod 139$ .

Обозначим зашифрованный код за  $\text{encode}(a, k)$ . Чтобы код можно было восстановить, хранится массив  $b$  длины  $n$ , в котором некоторый отрезок занят записанными подряд  $k$  и  $\text{encode}(a, k)$ . Например, если ключ равен  $[1, 2, 3]$ , а код равен  $[1, 5, 138, 12, 137, 60]$ , где-то в массиве  $b$  будет присутствовать отрезок

$[\dots, 1, 2, 3, 2, 7, 2, 13, 0, 63, \dots]$ .

Организаторы олимпиады справились восстановить доступ в систему, поэтому вам дается более простая задача. Вам дан массив  $b$ , а также уже восстановленный код  $a$  и длина ключа  $q$ . Найдите любое вхождение в  $b$  отрезка, удовлетворяющего условию.

### Формат входных данных

В первой строке ввода через пробел даны целые числа  $n$ ,  $m$  и  $q$  — длины имеющегося массива, кода и ключа, соответственно ( $1 \leq q \leq m \leq 10^6$ ;  $q + m \leq n \leq 10^6$ ). Гарантируется, что  $m$  делится на  $q$ .

Во второй строке через пробел перечислены  $m$  целых чисел  $a_i$  — элементы кода доступа  $a$  ( $0 \leq a_i < 139$ ). В третьей строке в том же формате перечислены  $n$  целых чисел  $b_i$  — элементы массива  $b$  ( $0 \leq b_i < 139$ ).

### Формат выходных данных

Выведите единственное целое число от 1 до  $n$  — позицию левой границы искомого отрезка массива  $b$ . Если существует несколько отрезков, удовлетворяющих условию, выведите левую границу любого из них.

Если данные некорректны, и удовлетворяющего условию отрезка не существует, выведите  $-1$ .

### Система оценки

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

Подзадача	Баллы	Дополнительные ограничения	Необходимые подзадачи	Информация о проверке
1	14	$n, m \leq 1000$		первая ошибка
2	11	$q = 1$		первая ошибка
3	20	$a_i = 0$ для всех $i$		первая ошибка
4	23	$q = m$		первая ошибка
5	32	нет	1 – 4	первая ошибка

## Примеры

стандартный ввод	стандартный вывод
12 6 3 1 5 138 12 137 60 0 5 1 2 3 2 7 2 13 0 63 44	3
18 6 3 1 2 1 2 4 2 72 71 72 0 0 0 1 2 1 2 4 2 3 6 3 4 8 4	4
5 2 2 1 1 1 2 3 4 5	-1