

Задача We are the Champions. Поеду домой

Имя входного файла:	input.txt или стандартный поток ввода
Имя выходного файла:	output.txt или стандартный поток вывода
Ограничение по времени:	2 секунды
Ограничение по памяти:	512 мегабайт

На третьем месяце дистанционного обучения студентке Насте стало скучно, и она решила поехать домой к себе в родной город на пару недель с целью развеяться. Чтобы ей было веселее в дороге, её друг подарил ей массив целых чисел a .

Шел пятый час поездки, и Настя вспомнила про подарок. Чтобы развлечь себя, она решила проверить, можно ли выбрать четыре **различных** индекса x, y, z, w таких, что $a_x + a_y = a_z + a_w$.

Занявшись этим делом, Настя не заметила, как прошло время, и её поезд уже пришел на станцию назначения. Она не успела закончить проверку, но всё ещё хочет узнать истину. Сможете ей помочь?

Формат входных данных

Первая строка содержит целое число n ($4 \leq n \leq 200\,000$) — количество чисел в массиве.

Вторая строка содержит целые числа a_1, a_2, \dots, a_n ($1 \leq a_i \leq 5 \cdot 10^6$).

Формат выходных данных

На первой строке выведите «YES», если четыре таких индекса существуют, и «NO» иначе.

Если такие индексы существуют, выведите в следующей строке x, y, z и w ($1 \leq x, y, z, w \leq n$).

Если возможных ответов несколько, выведите любой.

Примеры

ВВОД	ВЫВОД
6 2 1 5 2 7 4	YES 2 3 1 6
5 1 3 1 9 20	NO

Пояснение

В первом примере $a_2 + a_3 = 1 + 5 = 2 + 4 = a_1 + a_6$. Обратите внимание, что ответ 2 3 4 6 также подходит.

Во втором примере нельзя выбрать четыре таких индекса. Ответ 1 2 2 3 не подходит, потому что в нем не все индексы различны, хоть и $a_1 + a_2 = 1 + 3 = 3 + 1 = a_2 + a_3$.

Система оценки

Тесты к этой задаче состоят из трёх групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов всех предыдущих групп.

Группа	Баллы	Дополнительные ограничения	Комментарий
		n	
0	0	—	Тесты из условия.
1	30	$n \leq 60$	
2	30	$n \leq 600$	
3	40	—	

Задача This Is Gonna Hurt. Вставить текст

Имя входного файла:	input.txt или стандартный поток ввода
Имя выходного файла:	output.txt или стандартный поток вывода
Ограничение по времени:	6 секунд
Ограничение по памяти:	1024 мегабайта

Алиса и Аня работают копирайтерами. Недавно им пришёл заказ: нужно написать k текстов (строк) на схожую тематику. Девочки сразу приступили к работе и быстро получили k строк S_1, \dots, S_k , каждая из которых имеет длину не более m и состоит из строчных латинских букв, при чем длина S_1 оказался в точности равна m .

Девочки уже давно работают вместе, и у них есть простой способ проверить оригинальность своей работы. Оригинальность оценивается с помощью подстрок полученных текстов. Алиса и Аня считают некоторую строку неоригинальной, если и она, и строка, получающаяся из данной разворотом, встречаются в некоторых из их текстов в качестве подстрок на одних и тех же позициях.

Чтобы ускорить проверку, они разбивают строки на блоки. Разбиение на t блоков задается последовательностью a_0, a_1, \dots, a_t , где $a_0 = 0$, $a_t = m$ и $a_{i-1} < a_i$ для любого $1 \leq i \leq t$. Тогда i -м блоком называется отрезок целых чисел $[a_{i-1} + 1; a_i]$. Блок $[a_{i-1} + 1; a_i]$ называется интересным, если существует неоригинальная строка, встречающаяся в текстах ровно на позициях, задаваемых данным блоком. Иными словами, блок — интересный, если для каких-то строк S_l и S_r (возможно, $l = r$) верно, что $|S_l|, |S_r| \geq a_i$, и строка $S_{l, a_{i-1}+1} S_{l, a_{i-1}+2} \dots S_{l, a_i}$ совпадает со строкой $S_{r, a_i} S_{r, a_i-1} \dots S_{r, a_{i-1}+1}$, где $S_{t,j}$ — j -й слева символ строки S_t .

Например, для текстов [abba, ba] последовательности $(0, 1, 4)$ и $(0, 1, 2, 3, 4)$ задают корректные разбиения, а последовательности $(1, 2, 3)$ и $(0, 1, 1, 4)$ — нет. При этом для разбиения $(0, 2, 4)$ первый блок $[1; 2]$ является интересным, поскольку $S_{1,1} S_{1,2} = S_{2,2} S_{2,1} = ab$, а второй блок $[3; 4]$ — нет, поскольку для единственной возможной пары номеров $l = r = 1$ строки $S_{1,3} S_{1,4} = ba$ и $S_{1,4} S_{1,3} = ab$ не совпадают.

Разбиение называется интересным, если каждый блок в этом разбиении интересный. Алиса и Аня хотят найти интересное разбиение текстов на минимальное число блоков, чтобы измерить оригинальность работы. Девочки постарались, чтобы условие этой задачи было современным и прошло тест Бекдел, поэтому теперь помогите им и напишите для них программу, измеряющую оригинальность!

Заметим, что искомая величина корректно определена, так как, разбив строки на m блоков длины 1, мы получим интересное разбиение (в каждом блоке будет достаточно взять $l = r = 1$).

Формат входных данных

Во первой строке вводятся три целых числа t , k и m ($1 \leq k \leq 200\,000$, $1 \leq m \leq 500\,000$) — номер группы, к которой относится данный тест, общее число текстов и длина первого текста.

В i -й из следующих k строк вводится S_i — i -й текст, состоящий из строчных букв латинского алфавита.

Гарантируется, что $|S_1| = m$, $|S_i| \leq m$ для любого $i > 1$, и суммарная длина всех строк не превосходит 500 000.

Формат выходных данных

Во первой строке выведите единственное целое число t — минимальное число блоков в интересном разбиении.

Во второй строке через пробел выведите возрастающую последовательность из $t - 1$ целого числа a_1, \dots, a_{t-1} — номеров правых границ всех блоков, кроме последнего (само разбиение имеет вид $[1; a_1], [a_1 + 1; a_2], \dots, [a_{t-1} + 1; m]$).

Примеры

ВВОД	ВЫВОД
0 2 6 abcded cba	2 3
0 6 7 poggers sus amogus tokyo ghoul sodluv	4 3 5 6

Пояснение

В первом примере вторая строка `cba` при перевороте совпадает с первыми тремя символами первой строки `abcded`. Оставшиеся же символы `ded` образуют палиндром, т.е. эта строка совпадает с собой же перевёрнутой. Поэтому мы можем разбить строки на два блока $[1; 3]$ и $[4; 6]$. Легко видеть, что на меньшее число блоков разбить нельзя, ведь `abcded` — не палиндром.

Во втором примере в первом блоке $[1; 3]$ можно выбрать строку-палиндром `sus`, во втором блоке кусочки пятой (`ghoul`) и шестой (`sodluv`) строк, совпадающих друг с другом при перевороте. В третьем и четвертом блоках выбираем по одной букве из любой строки. Можно показать, что на меньшее число блоков разбить строки нельзя.

Система оценки

Тесты к этой задаче состоят из десяти групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов всех необходимых групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Обозначим суммарную длину всех строк как L .

Группа	Баллы	Доп. ограничения			Необх. группы	Комментарий
		k	m	L		
0	0	—	—	—	—	Тесты из условия.
1	7	$k \leq 100$	$m \leq 10$	$L \leq 1000$	0	
2	8	$k \leq 20$	$m \leq 100$	$L \leq 2000$	0	
3	10	$k \leq 200$	$m \leq 200$	$L \leq 40\,000$	0 – 2	
4	12	$k \leq 200$	$m \leq 2000$	$L \leq 400\,000$	0 – 3	
5	9	—	—	—	—	Гарантируется, что в ответе не больше двух блоков.
6	11	$k = 1$	$m \leq 50\,000$	$L \leq 50\,000$	—	
7	11	$k = 1$	—	—	6	
8	10	$k = 2$	—	—	—	
9	9	—	$m \leq 200\,000$	$L \leq 200\,000$	0 – 3, 6	
10	13	—	—	—	0 – 9	Offline-проверка.

Задача The Next Episode. Плитка для ванной

Имя входного файла: `input.txt` или стандартный поток ввода
Имя выходного файла: `output.txt` или стандартный поток вывода
Ограничение по времени: 5 секунд
Ограничение по памяти: 512 мегабайт

У Кости очень много дел — ремонт в самом разгаре! Надо клеить обои, собирать мебель и постоянно вывозить мусор.

Сегодня Костя хочет купить плитку для ванной. Он пришел в магазин и оказался перед большим квадратным стендом с плиткой. Стенд представляет из себя квадрат из $n \times n$ клеток, каждая клетка которого — маленький кусочек плитки цвета $c_{i,j}$. Магазин продает кусочки плитки пакетами — а именно, купить можно только подквадрат исходного квадрата.

Подквадратом называется любой квадратный фрагмент стенда, то есть любое множество $S(i_0, j_0, k) = \{c_{i,j} \mid i_0 \leq i < i_0 + k, j_0 \leq j < j_0 + k\}$ при $1 \leq i_0, j_0 \leq n - k + 1$.

Костя еще не знает, сколько кусочков плитки он хочет купить, и, соответственно, рассматривает подквадраты всех возможных размеров. При этом он точно не хочет слишком разноцветную плитку в ванной, что позволяет ему сузить выбор. Помогите Косте для каждого $k \leq n$ определить количество различных подквадратов размера $k \times k$, в которых не более q различных цветов плитки. Подквадраты считаются различными, если их расположение на стенде не совпадает.

Формат входных данных

В первой строке вводятся два целых положительных числа n, q ($1 \leq n \leq 1500, 1 \leq q \leq 10$) — размер стенда с плитками и ограничение на количество различных цветов в пакете.

В следующих n строках вводятся по n целых положительных чисел $c_{i,j}$ ($1 \leq c_{i,j} \leq n^2$) — j -е число в i -й строке соответствует цвету плитки в клетке (i, j) .

Формат выходных данных

Для каждого k от 1 до n выведите в отдельной строке по одному целому числу — количество подквадратов размера $k \times k$, которые интересны Косте.

Примеры

ВВОД	ВЫВОД
3 4 1 2 3 4 5 6 7 8 9	9 4 0
4 8 1 2 3 4 5 6 7 8 9 1 2 3 4 5 6 7	16 9 4 0

Пояснение

В первом примере все цвета квадратиков плитки различные. Поскольку Костя не хочет, чтобы в купленном квадрате было больше 4 цветов, он может купить себе любой подквадрат размера 1×1 или 2×2 , но при этом не сможет купить квадрат размера 3×3 .

Во втором примере есть повторяющиеся цвета. А именно, за счет ограничения $q = 8$ Костя может купить любой подквадрат 1×1 и 2×2 , а также любой подквадрат 3×3 , ведь в каждом таком подквадрате всего 7 цветов. Весь стенд размера 4×4 Костя купить не сможет, потому что там будет 9 цветов.

Система оценки

Тесты к этой задаче состоят из восьми групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов всех необходимых групп. **Offline-проверка** означает,

что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Дополнительные ограничения	Необх. группы	Комментарий
		n		
0	0	–	–	Тесты из условия.
1	5	$n \leq 10$	0	
2	6	$n \leq 50$	0–1	
3	7	$n \leq 200$	0–2	
4	13	$n \leq 500$	0–3	
5	14	–	0	Количество плиток каждого цвета не превышает 10.
6	15	–	0	$c_{i,j} \leq 20$
7	16	–	–	$q = 1$
8	24	–	0–7	Offline-проверка.

Задача Wind of Change. Фокус с подмножествами

Имя входного файла:	input.txt или стандартный поток ввода
Имя выходного файла:	output.txt или стандартный поток вывода
Ограничение по времени:	3 секунды
Ограничение по памяти:	512 мегабайт

Ваня придумал интересный фокус с множеством целых чисел.

Пусть у фокусника есть множество положительных целых чисел S . Он называет некоторое положительное целое число x . Зритель должен выбрать, не показывая фокуснику, некоторое подмножество S (возможно пустое). После этого зритель называет фокуснику размер выбранного подмножества. Фокус заключается в том, что после этого фокусник отгадывает: верно ли, что сумма элементов выбранного подмножества не превосходит x . Для пустого подмножества сумма предполагается равной 0.

Ване очень понравился этот фокус, поэтому он начал готовиться к тому, чтобы показать его публике. Для этого он приготовил некоторое множество **различных** положительных целых чисел S . Конечно, Ваня хочет, чтобы фокус обязательно получился. Он называет положительное целое число x *неудачным*, если не может быть точно уверен, что фокус пройдет удачно для любого подмножества, которое выберет зритель.

Чтобы оценить, насколько хорошее множество S он выбрал, он хочет посчитать количество неудачных положительных целых чисел для него.

Также Ваня планирует протестировать различные множества S . Поэтому он просит вас написать программу, которая найдет количество неудачных положительных целых чисел для изначального множества S и для множества S после каждого изменения. Ваня сделает q изменений своего множества, каждое изменение будет одного из двух видов:

- Добавить новое число a в множество S .
- Удалить некоторое число a из множества S .

Формат входных данных

В первой строке находятся два целых числа n, q ($1 \leq n, q \leq 200\,000$) — размер изначального множества S и количество изменений.

В следующей строке находятся n **различных** целых чисел s_1, s_2, \dots, s_n ($1 \leq s_i \leq 10^{13}$) — элементы изначального множества S .

В каждой из следующих q строк находятся два целых числа t_i, a_i ($1 \leq t_i \leq 2, 1 \leq a_i \leq 10^{13}$), описывающих очередное изменение.

- Если $t_i = 1$, то это операция добавления нового числа a_i в множество S . Гарантируется, что этого числа не было в множестве S до выполнения операции.
- Если $t_i = 2$, то это операция удаления числа a_i из множества S . Гарантируется, что это число было в множестве S до выполнения операции.

Формат выходных данных

Выведите $q + 1$ строку.

В первой строке выведите количество неудачных положительных целых чисел для изначального множества S . В следующих q строках выведите количество неудачных положительных чисел для множества S после каждого изменения.

Пример

ВВОД	ВЫВОД
3 11	4
1 2 3	1
2 1	6
1 5	12
1 6	19
1 7	13
2 6	8
2 2	2
2 3	10
1 10	3
2 5	0
2 7	0
2 10	

Пояснение

В первом тесте изначальное $S = \{1, 2, 3\}$. Для этого множества фокус может не получиться при $x \in \{1, 2, 3, 4\}$. Например, если $x = 4$, то зритель может загадать подмножество $\{1, 2\}$, сумма элементов которого равна $3 \leq x$, а может загадать подмножество $\{2, 3\}$, сумма элементов которого равна $5 > x$. Однако в обоих случаях зритель назовет фокуснику размер подмножества 2, поэтому он не сможет точно сделать правильный ответ. При этом поскольку подмножество размера 3 единственно, а сумма в любом подмножестве меньшего размера не превосходит 5, все $x \geq 5$ не являются неудачными.

Система оценки

Тесты к этой задаче состоят из семи групп. Баллы за каждую группу ставятся только при прохождении всех тестов группы и всех тестов всех необходимых групп. **Offline-проверка** означает, что результаты тестирования вашего решения на данной группе станут доступны только после окончания соревнования.

Группа	Баллы	Ограничения			Необх. группы	Комментарий
		n	q	Доп. ограничения		
0	0	—	—	—	—	Тесты из условия.
1	10	$n \leq 12$	$q \leq 100$	все $ S \leq 12$	0	
2	10	$n \leq 100$	$q \leq 100$	$s_i, a \leq 100$	0	
3	20	$n \leq 2000$	$q \leq 2000$	—	0 – 2	
4	15	$n \leq 50\,000$	$q \leq 50\,000$	—	0 – 3	
5	15	$n \leq 100\,000$	$q \leq 100\,000$	—	0 – 4	
6	15	$n \leq 150\,000$	$q \leq 150\,000$	—	0 – 5	Offline-проверка.
7	15	—	—	—	0 – 6	Offline-проверка.