# Сортировка матрицы

| | |
|---|---|
| Имя входного файла: | `стандартный ввод` |
| Имя выходного файла: | `стандартный вывод` |
| Ограничение по времени: | 2.5 секунд |
| Ограничение по памяти: | 256 мегабайт |

Даны две таблицы $A$ и $B$ размера $n \times m$.

Назовём *сортировкой по столбцу* следующее действие — выбирается столбец, и все строки упорядочиваются по значению в этом столбце, от строк, содержащих меньшие значения, к строкам с большими. В случае, если две строки имеют одинаковое значение в этом столбце, их порядок не меняется (такие сортировки называются *стабильными*).

Подобное поведение сортировки по столбцу можно найти практически в любом офисном приложении для работы с таблицами. Петя работает в одном из таких приложений, и у него открыта таблица $A$. Он хочет проделать ноль или более операций сортировки по столбцу, чтобы получить таблицу $B$.

Определите, может ли он это сделать, и если может, предложите ему алгоритм действий — последовательность столбцов, по которым нужно применить сортировку по столбцу. Обратите внимание, что **не требуется** минимизировать число действий.

## Формат входных данных

В первой строке даны два числа $n$ и $m$ ($1 \leqslant n, m \leqslant 2000$) — размеры таблицы.

Следующие $n$ строк содержат по $m$ значений $a_{i,j}$ ($1 \leqslant a_{i,j} \leqslant n$) в каждом и задают таблицу $A$.

Последние $n$ строк задают таблицу $B$ в том же формате, каждая из этих $n$ строк содержит по $m$ элементов $b_{i,j}$ ($1 \leqslant b_{i,j} \leqslant n$).

## Формат выходных данных

Если отсортировать таблицу $A$ в таблицу $B$ невозможно, выведите `-1`.

В противном случае выведите $k$ ($0 \leqslant k \leqslant 5000$) — количество сортировок, которые нужно сделать.

Затем выведите $k$ целых чисел $c_1, \ldots, c_k$ ($1 \leqslant c_i \leqslant m$) — номера столбцов, по которым нужно сделать сортировку.

Можно доказать, что если Петя может осуществить желаемое, то 5000 действий ему хватит.

## Примеры

| стандартный ввод | стандартный вывод |
|---|---|
| 2 2<br>2 2<br>1 2<br>1 2<br>2 2 | 1<br>1 |
| 3 3<br>2 3 2<br>1 3 3<br>1 1 2<br>1 1 2<br>1 3 3<br>2 3 2 | 2<br>1 2 |
| 2 2<br>1 1<br>2 1<br>2 1<br>1 1 | -1 |

## Замечание

Рассмотрим второй тест. После сортировки по первому столбцу таблица имеет вид

$$\begin{matrix} 1 & 3 & 3 \\ 1 & 1 & 2 \\ 2 & 3 & 2 \end{matrix}$$

После того, как мы отсортируем таблицу по второму столбцу, она станет

$$\begin{matrix} 1 & 1 & 2 \\ 1 & 3 & 3 \\ 2 & 3 & 2 \end{matrix}$$

что нам и нужно.

В третьем тесте любая сортировка не меняет таблицу, так как все столбцы уже отсортированы.

## Система оценки

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

Во второй группе — любой столбец таблицы является перестановкой чисел от 1 до $n$.

| Группа | Баллы | Дополнительные ограничения | | Необх. группы | Комментарий |
|---|---|---|---|---|---|
| | | n | m | | |
| 0 | 0 | — | — | — | Тесты из условия |
| 1 | 6 | $n \leqslant 6$ | $m \leqslant 6$ | 0 | |
| 2 | 4 | — | — | — | Все столбцы в $A$ и $B$ являются перестановками. |
| 3 | 16 | $n \leqslant 500$ | $m \leqslant 6$ | 0, 1 | |
| 4 | 10 | $n \leqslant 100$ | $m \leqslant 100$ | 0, 1 | |
| 5 | 15 | $n \leqslant 500$ | $m \leqslant 500$ | 0, 1, 3, 4 | |
| 6 | 18 | $n \leqslant 1000$ | $m \leqslant 1000$ | 0, 1, 3, 4, 5 | |
| 7 | 31 | $n \leqslant 2000$ | $m \leqslant 2000$ | 0 – 6 | **Offline-проверка.** |