B: Acceptable Seating Arrangements

Time Limit: 1 second, Memory limit: 2G

Charlie is managing a classroom. The seats in the classroom are arranged in a grid with rows and columns. Each student has a distinct height.

A configuration of students to seats is *acceptable* if the following conditions are met:

- Each student is assigned to exactly one seat.
- The students are seated in increasing order of height from left to right in each row.

The students are initially seated in an acceptable arrangement. Charlie wants to rearrange students into a potentially different acceptable arrangement. To do this, he can swap any two students. However, he wants to ensure that the configuration stays acceptable after each swap.

Help Charlie devise a strategy to move the students from the original arrangement to his preferred arrangement. You don't need to minimize the number of swaps, but you are limited to at most 10^4 swaps.

It can be proven that this is always possible for all possible inputs that satisfy the input constraints.

Input

The first line of input contains two integers r and c ($1 \leq r, c \leq 20$). Charlie's classroom has r rows and c columns of seats.

Each of the next r lines contains c integers h ($1 \leq h \leq r \cdot c$), representing the heights of the students in each row in the original arrangement. The heights are guaranteed to be distinct, and the arrangement is guaranteed to be acceptable.

Each of the next r lines contains c integers h ($1 \leq h \leq r \cdot c$), representing the heights of the students in each row in Charlie's desired arrangement. The heights are guaranteed to be distinct, and the arrangement is guaranteed to be acceptable.

Output

On the first line, output an integer k, which is the number of swaps to perform ($0 \leq k \leq 10^4$).

Then, output the k swaps which change the original arrangement to Charlie's preferred arrangement. On each of the next k subsequent lines, output four integers r_1, c_1, r_2, c_2 ($1 \leq r_1, r_2 \leq r$, $1 \leq c_1, c_2 \leq c$, $(r_1, c_1) \neq (r_2, c_2)$). This represents a swap of the student in row r_1, column c_1 with the student in row r_2, column c_2.

It can be proven that it's always possible to accomplish this in under 10^4 swaps for all possible inputs that satisfy the input constraints. Remember that the arrangement must be acceptable after each swap.

Sample Input 1	Sample Output 1
2 3 1 4 5 2 3 6 3 5 6 1 2 4	4 2 1 1 1 2 2 1 1 2 3 1 3 2 3 1 2