Problem 3

The Tower of Hanoi is a popular puzzle. There are three pegs, and a number of discs of different sizes which can slide onto any peg. The puzzle starts with the discs initially stacked in order of size on 1st peg, smallest at the top. The object of the game is to move the entire stack to 2nd peg using the 3rd as auxiliary one, obeying the following rules:

[1] Only one disc may be moved at a time.
[2] No disc may be placed on top of a smaller disc.

A straightforward recursive solution is to first move the top (n-1) peg recursively to the auxiliary peg using the destination peg. Then shift the n-th peg to destination in one move, and then again recursively move the (n-1) pegs to the destination peg.

Your task is to obtain one configuration from another in minimum number of moves. A configuration of n pegs is specified by a sequence of n integers, with values either 1, 2 or 3, denoting the location (peg number) of the ***i***-th disk. There is a unique sequence with minimum number of moves.

Input
There are several test cases. Each test case starts with an integer n (1 <= n <= 50). Then the next two lines give the source and destination configuration respectively. Each configuration is specified by a sequence of n integers (1, 2 or 3) separated by space.
A zero (0) denotes end of input. This should not be processed.

Output:
For each test-case print the sequence of moves, one move per line. Each move should start with the disk id (between 1 to n), a colon, then the source and destination peg-ids. Separate outputs of different test-cases by a blank line.

Sample Input:

```
4
1 2 3 1
3 3 3 2
5
1 2 2 2 2
3 2 2 2 2
0
```

Sample Output:

```
2 : 2 3
1 : 1 3
4 : 1 2

1 : 1 3
```