Hanoi Calls

Theory:

- Towers of Hanoi is an arrangement consisting of three pegs and **N** discs of radius 1 to **N**.
- Each peg can hold zero or more discs, but at any point of time, the radius of the discs must be in decreasing order from bottom to top.
- A move consists of moving the topmost disc from one peg to another. After the move, the descending order property of pegs must hold.

Traditional problem is: If all discs are stacked up on peg#1, how many moves will it take to move all the discs to peg#2?

Recursive solution: Noting that for disc N to move, from peg #a to peg #b, all discs of size 1 to N-1 must be in peg #c. Hence there is exactly one minimal way to move the discs. After disc N has moved, all pegs from #c must be moved back to #a. If moves(N) denote the number of moves required to transfer N discs between two pegs (both sorted configuration), then moves(N) = moves(N-1) + 1 + moves(N-1); Solving the recurrence yields moves(N) = 2^N -1; The idea i am trying to share is that, there is exactly one such move sequence.

Now the problem is that given any initial configuration of the discs, and any final configuration, Can you tell me the minimal number of moves required to change it from initial to final configuration?

Input Format:

The input file consists of multiple testcases.

The first line of each testcase contains one integer, **N** (1 <= N <= 30)

The second line of each testcase contains N integers, each one of which will be between 1 and 3. The **i**-th integer tells you the peg number at which disc of radius **i** is present in the initial configuration.

The third line contains a similar specification for the final configuration.

Input terminates with a line containing a single zero, which must not be processed.

Output Format:

For each testcase print one line containing a single integer, which is the minimal number of moves to make the transfer.

Test data:

100 testcases

Sample Input:

4
1 1 1 1
2 2 2 2
3
1 3 3
2 1 1
5
1 3 2 2 2

2 3 2 1 2
0

Sample Output:

15
6
14

Output Explanation:
TestCase#1:
This is the moves(4) = 2^4 -1;
TestCase#2:

  [peg1, peg2, peg3] =
#0 [{1}, {}, {3,2}]
#1 [{1}, {2}, {3}]
#2 [{}, {2,1}, {3}]
#3 [{3}, {2,1}, {}]
#4 [{3}, {2}, {1}]
#5 [{3,2}, {}, {1}]
#6 [{3,2}, {1}, {}]